Population-Based Algorithms for Improved History Matching and Uncertainty Quantification of Petroleum Reservoirs

Yasin Hajizadeh

Submitted for the Degree of Doctor of Philosophy

Institute of Petroleum Engineering
Heriot Watt University
Edinburgh
UK

March 2011

Abstract

In modern field management practices, there are two important steps that shed light on a multimillion dollar investment. The first step is history matching where the simulation model is calibrated to reproduce the historical observations from the field. In this inverse problem, different geological and petrophysical properties may provide equally good history matches. Such diverse models are likely to show different production behaviors in future. This ties the history matching with the second step, uncertainty quantification of predictions. Multiple history matched models are essential for a realistic uncertainty estimate of the future field behavior. These two steps facilitate decision making and have a direct impact on technical and financial performance of oil and gas companies.

Population-based optimization algorithms have been recently enjoyed growing popularity for solving engineering problems. Population-based systems work with a group of individuals that cooperate and communicate to accomplish a task that is normally beyond the capabilities of each individual. These individuals are deployed with the aim to solve the problem with maximum efficiency.

This thesis introduces the application of two novel population-based algorithms for history matching and uncertainty quantification of petroleum reservoir models. Ant colony optimization and differential evolution algorithms are used to search the space of parameters to find multiple history matched models and, using a Bayesian framework, the posterior probability of the models are evaluated for prediction of reservoir performance.

It is demonstrated that by bringing latest developments in computer science such as ant colony, differential evolution and multiobjective optimization, we can improve the history matching and uncertainty quantification frameworks. This thesis provides insights into performance of these algorithms in history matching and prediction and develops an understanding of their tuning parameters. The research also brings a comparative study of these methods with a benchmark technique called Neighbourhood Algorithms. This comparison reveals the superiority of the proposed methodologies in various areas such as computational efficiency and match quality.

تقدیم به

دو عشق پاک زندگی:

پدر بزرگوار و مادر عزیزم

Dedicated to two pure loves of life: My kind father and dear mother

Acknowledgements

“Yasin
I have had a quick look at your email. Based on a quick read, you have made it past the "immediate reject" barrier, so well done...!”

This was the first reply (and one of the most pleasant ones) I received from Mike to my initial ideas to apply ant colony optimization in petroleum engineering. Having a series of emails and an interview, I eventually received an offer of PhD in the uncertainty quantification group. After this, Mike was always there to help me in passing other barriers. Mike has been a wonderful adviser and changed the way I look at everything, including science. Thank you Mike for giving me this opportunity and all the support you had during my PhD journey. Thank you for leaving me alone; so I could learn how to work independently and be more productive. It was a great pleasure working with you. I would also like to express my thanks to Vasily for the fruitful discussions we had around various topics and also bringing some of the best chocolates I ever had to our office. They made my life in the uncertainty group sweeter.

Special thanks go to the uncertainty project sponsors BP, BG, ConocoPhillips and JOGMEC for their financial support during my PhD studies. Eclipse software donation from Schlumberger is also gratefully acknowledged here.

This is a good opportunity to thank the computer support team for their help in making computations smoother. I would also like to thank the secretarial and purchase teams for their continuous support.

Thank you to Mohammad, Asaad, Simon, Lorna, Dan, Jeb, Houra, Panteha, Hossein, Masoud, Farzaneh, Alireza, Hamidreza, Amir, Mahmood, Reza, Hoda, Manooch, Hooman, Mehran, Mohsen and all other friends who made my stay at Heriot Watt University and Edinburgh an enjoyable and memorable experience.

I greatly appreciate the invaluable comments from my examiners, Dr Muggeridge and Prof Corne, which significantly improved the quality of my work.

Last but not least, my heartfelt gratitude goes to my lovely parents for their endless support during my life and while I was away from home for my PhD. Azar, the angel of my life, sacrificed her life for me and Hasan, the hero of my life, taught me how to measure things in my real life. I love you!

Publications

Book Chapter:

1. Hajizadeh, Y., Demyanov, V., Mohamed, L., Christie, M [2010] Comparison of Evolutionary and Swarm Intelligence Methods for History Matching and Uncertainty Quantification in Petroleum Reservoir Models, In: Koeppen et al. (Eds.) Intelligent Computational Optimization in Engineering: Techniques and Applications, Springer

Journal Paper:

Hajizadeh, Y., Christie, M., Demyanov, V [2011] Ant Colony Optimization for History Matching and Uncertainty Quantification of Petroleum Reservoir Models, Submitted to Journal of Petroleum Science and Engineering (In Press)

Conference Papers:

1. Hajizadeh, Y., Christie, M., Demyanov, V [2011] Towards Multiobjective History Matching: Faster Convergence and Uncertainty Quantification, SPE 141111, Reservoir Simulation Symposium, The Woodlands, Texas, USA, 21-23 February
2. Hajizadeh. Y, Christie. M, Demyanov. V [2010] Comparative Study of Novel Population-based Optimization Algorithms for History Matching and Uncertainty Quantification: PUNQ-S3 Revisited, SPE 136861, ADIPEC Conference, Abu Dhabi, United Arab Emirates, 1-4 November
3. Hajizadeh, Y., Christie, M., Demyanov, V [2010] History Matching with Differential Evolution Approach; A Look at New Search Strategies, SPE 130253, SPE EUROPEC Conference, Barcelona, Spain, 14-17 June
4. Hajizadeh, Y [2010] On the Inverse Problem of History Matching in Petroleum Engineering, Stochastic Population-Based Optimization Viewpoint, Inverse Problems: Modeling and Simulation, Antalya, Turkey, 24-29 May
5. Hajizadeh, Y., Christie, M., Demyanov, V [2010] The Value of Production Data in History Matching and Uncertainty Quantification Using Evolutionary Optimization Algorithms, 6th IMA Conference on Modeling Permeable Rocks, Edinburgh, UK, 29 March-1 April

6. Hajizadeh, Y., Christie, M., Demyanov, V [2009] Application of Differential Evolution as a New Method for History Matching. SPE 127251, SPE International Petroleum Conference, Kuwait, 14-16 December
7. Hajizadeh, Y., Christie, M., Demyanov, V [2009] Ant Colony Optimization Algorithm for History Matching. SPE 121193, SPE EUROPEC Conference, Amsterdam, The Netherlands, 8-11 June

Contests:

Ants Can Do History Matching: Presented in the SPE Student Paper Contest Series in PhD Division

First Place Winner of European Regional Contest, Offshore Europe Conference, Aberdeen, UK, 2009

Second Place Winner of International Student Paper Contest, SPE Annual Technical Conference and Exhibition, Florence, Italy, 2010

Table of Contents

List of Figures

Chapter 1

Chapter 2

Chapter 3

Chapter 4

Chapter 5

Chapter 6

Chapter 7

List of Tables

Chapter 4

Chapter 5

Chapter 6

Chapter 1

> "Persons pretending to forecast the future shall be considered disorderly under subdivision 3, section 901 of the criminal code and liable to a fine of $250 and/or six months in prison."
>
> Section 889, New York State of Criminal Procedure

Introduction

1.1 What is the Value?

Fossil fuels will remain the dominant source of energy through the middle of this century even under the most optimistic assumptions about the development of alternative technologies. According to the world energy outlook published by International Energy Agency (IEA) [2008], the world's primary energy demand will grow by 1.6% per year on average between 2006 and 2030. IEA predicts that oil will remain the largest single fuel in the global energy mix through 2030 although its demand share will drop. Global oil demand is predicted to reach 99 million barrels per day (MB/D) in 2015 and 106 MB/D in 2030 - up from 85 MB/D in 2008.

With this ever growing need for energy, the oil industry is trying to respond to this demand with new exploration programs and enhancing production from existing fields. This, of course, needs massive investments. IEA expects that around 12 trillion dollars

will be spent in the oil and gas sectors in 2007-2030. Figure 1 shows the distribution of these costs for both oil and gas businesses.

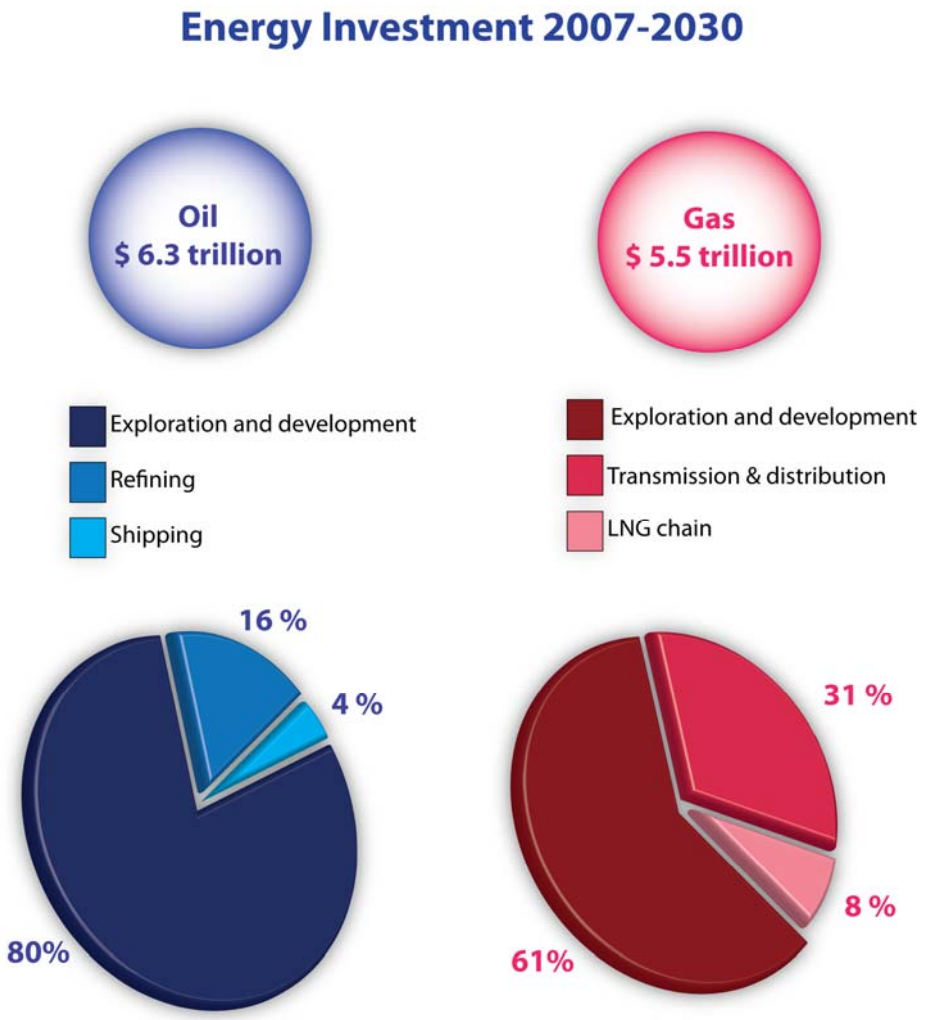

Figure 1: Energy investment in 2007-2030 and distribution of this investment for different sectors (numbers from IEA report [2008])

1.2 It is All About Improved Decision Making!

Making correct investment decisions depends on our knowledge about a field and its future performance. The main objective is to optimize the value of a project, an asset or a reservoir. There are many modern sophisticated tools to assist subsurface teams in their decision making. All of these tools have one thing in common and that is the requirement to understand the underground reservoir and the ability to predict its future performance.

During the last few years, “digital field” solutions have been developed and deployed in most major oil companies in order to facilitate the field management process [Sankaran et al. 2009]. In a digital oil field we deal with the intensive use of automation and information technologies in order to measure, model and control the assets with an integrated approach to make real time decisions, resulting in an increased ultimate recovery. This framework has been implemented successfully in major oil companies including BP (field of the future) [Thomson, 2008], Shell (smart fields) [Best and den Berg, 2006] and Chevron (i-field) [Ouimette and Oran, 2006].

We should not also forget the need to enhance production from our mature assets and the importance of modern field studies and improved decision making in this process. Large volumes of oil remain in many old fields despite the fact that some of them have been producing for decades. Statistics show that the number of field re-development projects is increasing [Heward and Gluyas, 2002]. The key factor for having a successful re-development operation is a proper field study which incorporates previous information, gathers new data, analyzes them and uses them to make decisions.

Within the sophisticated digital fields context or traditional management procedures, either for developing a new field or rehabilitation projects of mature fields, a general framework is used within oil and gas companies for field studies and decision making processes. It is all about improved decision making!

1.3 Field Management Workflow

Several workflows have been developed both in academia and the industry for describing the steps taken in a field management practice. Integration of subsurface, surface and economic analysis is often seen at the heart of modern field planning and operations [Serbini et al. 2009]. New technologies such as interactive web-based systems to setup, support and monitor field studies have helped to ease the process [Volpi et al. 2008]. Based on recent advances, a general workflow is presented in figure 2 for an integrated field study.

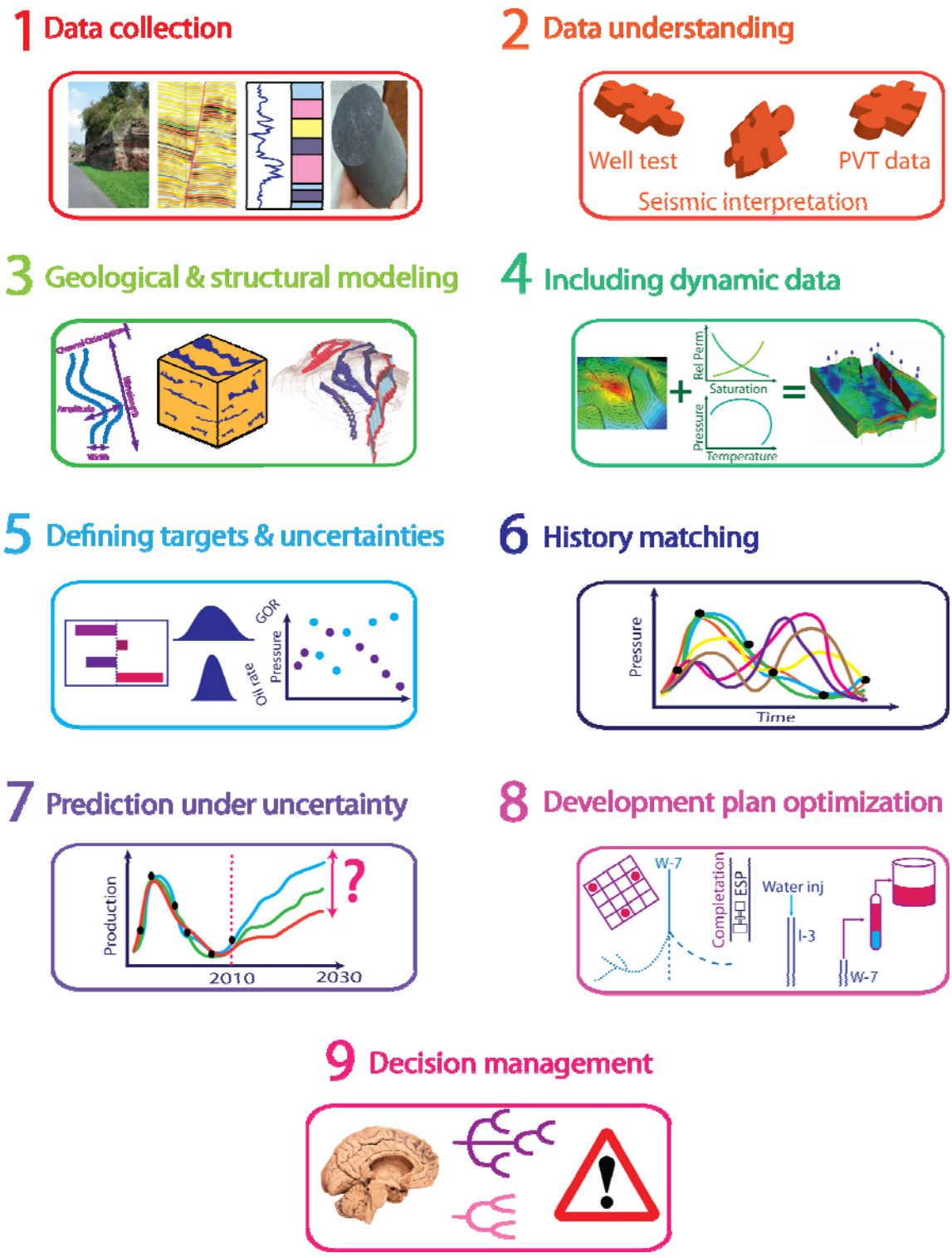

Figure 2: General field study and development workflow

1.3.1 Data Collection

Integrated data acquisition is the first step in understanding and modeling of a reservoir. This step should address the data needs, application areas and the cost/benefit

assessment regarding collecting the required data. We are fortunate that today we are able to obtain real time information from our reservoirs. For example during drilling of wells, measurement, logging and seismic while drilling (MWD/LWD/SMD) tools can be used to obtain real time information about reservoir properties [Anchliya, 2006], [Zhou and Mardambek, 2008]. Some very important decisions must be made at the data collection stage. For example the number of appraisal wells and their locations must be carefully selected to represent the variations in reservoir quality and fluid properties. Examples of data that can be collected in this step include outcrop studies, 2D and 3D seismic, core, fluid properties, well log and well tests. Further to these traditional data, recently some new technologies such as satellite images or Interferometric Synthetic Aperture Radar (InSAR) are being used for monitoring subsurface fluid flow and pressure change [Du et al. 2010]. Using such advanced technologies besides other sources of reservoir information results in having a more reliable data set which in turn is the foundation of a successful field study.

1.3.2 Understanding, Integration and Interpretation of Data

After the data has been acquired, it must be carefully evaluated. We should check data quality to ensure that it is suitable for achieving the objectives of the project. It has been shown that a well-targeted data collection plus rigorous data analysis help to gain a better understating of reservoirs [Abu El Ela, 2007]. For example, having sufficient and representative reservoir fluid samples is an important stage for obtaining reliable PVT data; but also understanding the data quality and evaluating the consistency of data is essential in this process [Lawrence and Gupta, 2009]. Developing a solid understanding and integration of the data helps to have a common insight to the reservoir. Modern data analysis and visualization tools [Sultanum et al. 2010] are necessary in this step. Data mining techniques can be used to discover patterns and relationships that are repeated in the data. Advanced data filtering techniques and noise reduction algorithms are also necessary for correct interpretation of some data types; for example seismic data [Eisenberg-Klein et al. 2008], [Elboth et al. 2009]. The results of step 2 play a crucial role in the field management workflow and will be later incorporated in construction of the reservoir simulation model.

1.3.3 Building Structural and Geological Model

In this step, and based on the dat obtained, a structural and geological model of the reservoir must be constructed. Such a model is developed by analyzing available seismic and sparse well log data. The structural model represents major faults and horizons and is considered as a stage where the reservoir-scale structure and data will be embedded. At this step reservoir boundary and segments, zonation and layering systems are defined in the model. Different logs and borehole images can help in building the structural model. With recent advances in automatic fault network connectivity detection and stratigraphic sequence modeling, better analysis of fault geometry and displacement patterns is possible and a structural model can be built more accurately.

The static geological model aims to predict the distribution of reservoir facies throughout the 3D volume of the reservoir. After calculating their distribution, facies are populated using object-based (Boolean) modeling or other techniques. Next, petrophysical properties of cells are included in the model following a geostatistical approach (sequential Gaussian simulation) or similar methods.

1.3.4 Adding Dynamic Data and Completing Flow Simulation Model

Including dynamic reservoir and fluid properties is an important task in building the digital reservoir model. At this stage, the dynamic properties of the model are added by the reservoir engineering team. Examples of data added to the simulation model at this step include relative permeability curves, fluid PVT data, aquifer properties, etc. Usually the number of original grid cells generated during geological modeling is too big for handling in today's flow simulation softwares. Thus, the fine-scale geological model is upscaled at this stage to reduce the number of grid cells and make a model suitable for running a flow simulation. However this may rapidly change in the future. For example Saudi Aramco has developed the GIGAPOWERS simulator [Dogru et al. 2009] which is capable of handling billion-active-cell models. This step completes the reservoir model and makes it ready for flow simulation.

1.3.5 Defining Key Uncertainties and Goals of History Matching

Before launching a history matching study, a clear list of targets must be developed. This list should address our expectations from history matching and the target variables

to obtain a match. Another key issue that should be agreed within the team is the definition and level of an *acceptable* match.

After defining key targets, a preliminary sensitivity study must be performed to determine the uncertain variables and assess their impact on the selected match targets. The outcome of this stage is a list of variables that must be modified in the history matching process and their prior range.

One of the other main issues at this stage is the amount of data. Hundreds of data points (for example production rate measurements) might be available for a mature field. Outliers in data may result in obtaining unrealistic misfit values in history matching. Thus, a rigorous data analysis is also necessary in this step to filter available data and select the appropriate measurement points to be used in history matching.

1.3.6 History Matching

History matching is a process where the current reservoir simulation model is conditioned to available field data. It aims to tune the model in order to be consistent with the field performance. A simulation model which can capture the past life of a reservoir is more likely to make accurate predictions. History matching also acts as a way of validating other data. For example pressure analysis or well performance data may indicate the possibility of a fault; however seismic data may not confirm the suggestion of previous data. Here, the presence of a fault can be inserted as an uncertainty into the history matching framework and then history matching will provide more evidence to this issue.

History matching is an ill-posed inverse problem with non-unique solutions. Multiple realizations of the reservoir may give equally good matches to available data. We demonstrate this in figure 3. In this simple history matching problem permeability values in three layers are adjusted to obtain a match to measured production rates. In the left side of figure 3 we have the so called search space where we have tried 7 different combinations of permeability values. In the right side of figure 3 we present the resulting behavior of the reservoir and its match to oil rate measurements.

Out of 7 cases tried for this problem, 3 models (A, F and G) resulted in an acceptable match. Imagine we want to have a waterflooding project in this reservoir. Zones with high permeability values may cause early water breakthrough in the oil producers and uneven sweep around water injectors [Al-Dhafeeri and Nasr-El-Din, 2007]. Misidentification of the layers can ultimately result in an unsuccessful waterflooding project. This example clearly shows the importance of finding multiple realizations in history matching studies and its impact on reservoir management practices.

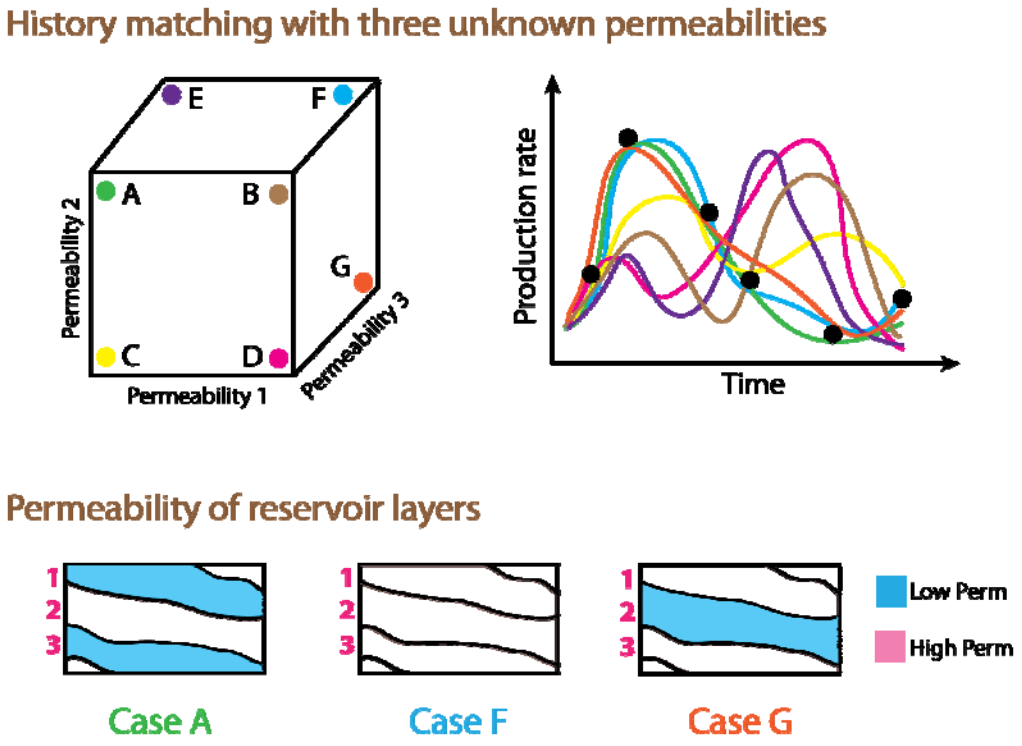

Figure 3: History matching of a simple reservoir model with three unknown permeability values and the multiple history-matched models with different permeability in layers

Real life history matching problems are restricted by the number of simulations that can be performed in a reasonable time. Running a single simulation may take hours or even days in a real life problem. This limitation motivates the development of novel optimization algorithms to navigate the search space and find multiple models. This navigation of must be fast (limited number of simulations) and efficient (identify multiple history-matched realizations).

Effective history matching is the first focus area of this thesis. It is hoped that the algorithms proposed in this thesis will be more efficient in navigating the search space

and finding multiple history-matched models from fewer simulations, thus saving computational resources.

1.3.7 Predicting Performance of the Models under Uncertainty

Over the years our industry has moved from “in data we trust” to “in uncertainty we trust”. One of the main concerns in reservoir engineering studies is to get reliable production forecasts to make optimal management decisions both from technical and economical viewpoints.

The ultimate goal of history matching is to have a calibrated reservoir model with high prediction capability. The traditional, yet common, way of making predictions of future recovery is running a single best history-matched model to forecast a period of time. Due to the non-uniqueness of the solution, the obtained recovery estimate is uncertain and might be far away from reality. However this view is starting to change within the oil industry by defining multiple scenarios and using multiple reservoir models to perform field development studies.

The industry focus is shifting to have reliable uncertainty estimates through multiple scenarios/realizations of the reservoir. This stage of the reservoir management will be the second focus area of this thesis. We will examine the uncertainty of predictions made by multiple history matched models and how the uncertainty estimates are affected by the choice of optimization algorithms and their tuning.

1.3.8 Development Plan Optimization

After obtaining calibrated reservoir models, the field development plan must be optimized. In this step all economic and technical targets are listed and development strategies are defined. Well placement in the reservoir, well type (vertical, horizontal and multi lateral), well spacing, assigning maximum rates and artificial lift strategies are some of the issues covered at this stage. Other challenges at this stage are design of enhanced oil recovery strategies like water flooding or gas injection. In designing development plans, the integration of subsurface and surface facilities should be considered as the key point [James et al. 2008].

Development plan optimization will heavily depend on the field constraints and/or company policy. We should take into account some critical factors like drilling rigs, available gas for injection in future and other technical, economical or even political factors in order to optimize current plans.

1.3.9 Decision Management

Most of the decisions in our industry are made under uncertainty with limited information about their consequences. For a long time decision making under risk and uncertainty has fascinated observers of human behavior. Recent studies on the brain using functional magnetic resonance imaging (fMRI) have shown significant changes in brain activity while making a decision under uncertainty [Huettel et al. 2005]. This is related to the approach we take in facing dynamic uncertainty and learning from previous experiments and regrets. In fact, anticipating regret is a powerful predictor of future choices and suggests that decisions can be shaped by human emotions [Coricelli et al. 2005].

Decision management can't be considered as a separate stage in field development workflow, but it is believed that decision management must be coupled with all other steps. Other steps in the workflow provide the management team with technical and economical information in order to make the final decisions. Decision making under uncertainty is an essential component of everyday life, so in all steps of the field management workflow. In fact, it is the decision that dictates what kind of approach must be taken in any field development project.

There are two separate approaches for decision making under uncertainty. We can have a *decision from experience* where we learn from the feedback provided by repeated sampling of available choice alternatives by personally doing a trial-and-error; or we can make a *decision from description* where we are provided with external provision of a numeric or graphic probability distribution.

No matter which approach we choose, there will still remain critical stages which require a human decision maker. For example, let us consider investigating the profitability of drilling an infill well in our reservoir. For this purpose, the impact of the new well on the production rate of the reservoir must be studied using the simulation

model. Looking to answer this question, Erbas [2006] presented three reservoir models with the same parameterization and uncertain variables, each history-matched using a different optimization algorithm. The result, as shown in figure 4, reveals that the history-matched model using the genetic algorithm (GA) rejects the profitability of drilling an infill well due to an economic limit, while the neighbourhood algorithm (NA) approves drilling the well. The critical step here is making the choice whether to drill the well or not.

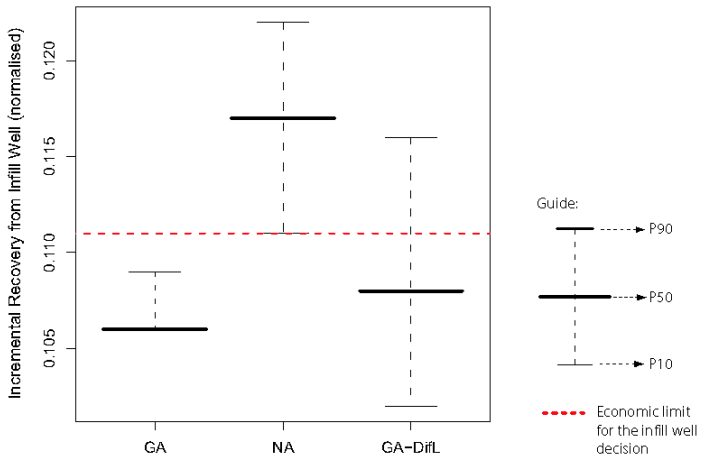

Figure 4: Three different optimization methods for assessing feasibility of drilling an infill well and their results considering the economic limit (taken from Erbas [2006])

Also, very often, new information acquired in a field may result in redefining reservoir architecture, thus changing the initial development plan. For example in the Clair field [Witt et al. 2010] new ocean-bottom seismic data, acquired 4 years after submitting the initial development plan, resulted in a new interpretation of fault positions and fracture orientations. In a response to the new interpretation, the development plans have been revisited. The process of updating old decisions requires advanced engineering tools coupled with real-time portfolio analysis which needs to be adopted within oil companies.

An important factor in improving current decision management practices is studying the psychological aspects of decision making in presence of uncertainty. Recently some work has been done in this area [Mackie et al. 2008] [Welsh et al. 2009]. The author believes this process should be dynamic and there must be a scope for updating the

decision by observing the outcomes. For this purpose, having multiple alternatives is essential at this stage. These improvements require much further work to overcome current barriers in accepting modern decision making concepts in our industry.

1.4 Who are the Customers of This Thesis?

This thesis focuses on steps 6 and 7 of the field management workflow (i.e. history matching and uncertainty quantification of predictions). This research aims to contribute to these steps by investigating two new optimization algorithms for assisted history matching. The customers of this thesis are reservoir management teams who want to effectively update their reservoir simulation models, obtain multiple realizations of the reservoir that match to their production and other field performance data and quantify the uncertainty of future production. These steps have a crucial impact on the technical and financial performance of the companies. It is hoped that the proposed algorithms will enhance the performance of current assisted history matching frameworks. Faster convergence, more diverse models and accurate uncertainty quantification form back-bones of this research. Although we have focused on history matching in this thesis, the proposed algorithms can also be used to optimize other aspects of field management workflows such as well location, reservoir production and economic optimization.

1.5 Objectives of the Thesis

In this research we aim to:

- Apply ant colony optimization (ACO) and differential evolution (DE) to the history matching problem
- Compare the performance of these new algorithms with the neighbourhood algorithm
- Study the effect of tuning parameters on the performance of the algorithms and test their sensitivity to initial starting points
- Understand the influence of production data used for history matching on the uncertainty estimates
- Extend ACO and DE algorithms for handling multiple objectives in history matching problems and examine the effect of multiobjective history matching on uncertainty estimates

1.6 Thesis Preview

Chapter 2 reviews different techniques used for history matching of reservoir models and uncertainty quantification of predictions during the past 50 years.

Chapter 3 is devoted to a literature review and description of the working mechanisms of ant colony optimization, differential evolution, the neighbourhood algorithm and NAB.

Chapter 4 describes the application of ant colony optimization, differential evolution and the neighbourhood algorithms to the history matching of the Teal South reservoir. The Teal South case is a real reservoir with simple structure and a single producing well which is used as a proof-of-concept example. Performance of algorithms, convergence behavior and their sensitivity to the initial seed will be studied in this chapter. Uncertainty of predictions made by an ensemble of history-matched models will also be discussed. To demonstrate the value of information, the effect of data points used in history matching on the uncertainty estimates are also examined in this chapter.

In chapter 5, a comparative study of the proposed algorithms for history matching and uncertainty quantification of the PUNQ-S3 model is presented. The PUNQ-S3 model has a more complex geological structure than the Teal South model, which entails solving a high dimensional optimization problem. This model is fitted to multivariate production data coming from multiple wells.

Chapter 6 is dedicated to the extension of ant colony optimization and differential evolution algorithms for multiobjective optimization. Proposed mechanisms for this purpose are introduced and tested for a benchmark function suite. In the second part of this chapter, multiobjective history matching is tested on the PUNQ-S3 reservoir. The performance of multiobjective algorithms for history matching and uncertainty quantification is compared with the traditional objective-sum approach.

Chapter 7 presents the summary of research, major contributions and key conclusions. Recommendations for future work are also provided in this chapter.

Chapter 2

> "I don't know which one makes a man more conservative – to know nothing but the present, or nothing but the past"
>
> John Maynard, The End of Laissez-faire (1926)

History Matching and Uncertainty Quantification of Reservoir Models

Imagine as a member of a subsurface team you have been given some information about a field and have been asked to make a decision about improved production strategies. Using sophisticated tools, you will launch a field study. Having field production/seismic data, you may perform history matching and update the reservoir model. Then you will run a forecast simulation to understand the performance of the field in the future. Finally based on your study, you will propose a strategy to enhance the current production from the field.

When it comes to making decisions around reservoir management, there is only one thing that we are certain about; uncertainty is a certainty. The certainty is the incorrectness of the model used in the above procedure. It is very likely that you will receive a different response from the field in future than the one which has been predicted by your study.

This chapter will review various methods used for history matching. Sources of uncertainty in reservoir engineering and different methods for handling these uncertainties will be also covered in this chapter.

2.1 History Matching

As briefly introduced in chapter 1, history matching is an important step in any reservoir engineering study. During this stage, the initial reservoir simulation model is updated based on dynamic information coming from field performance. This information may include production data, tracer data, seismic studies, etc.

History matching is a process in which reservoir parameters are changed until the responses of computer simulations closely match available historical field data. In a history matching study, we aim to gain more information about the uncertain parameters. The ultimate goal is to update the reservoir model in such a way that it becomes a suitable tool for future predictions. Figure 1 presents a typical history matching workflow.

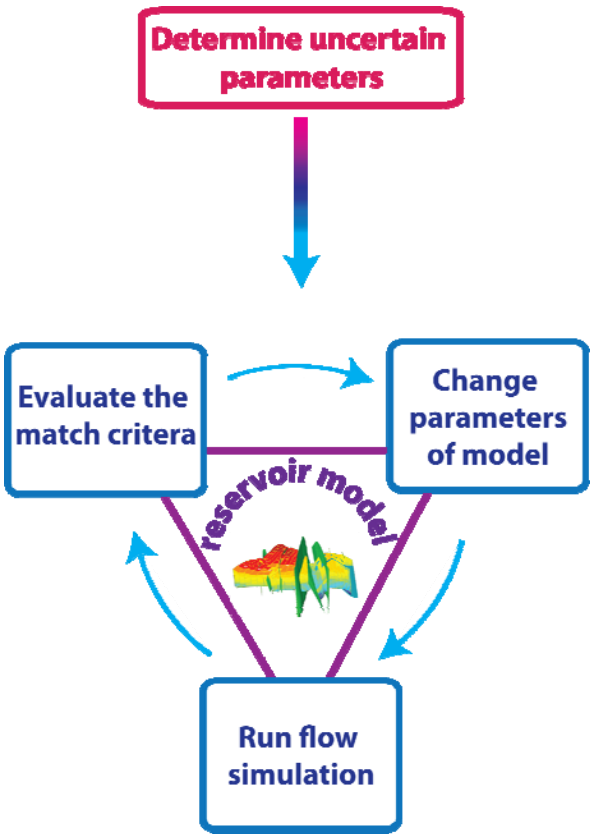

Figure 1: A typical history matching workflow

2.1.1 History of History Matching

Automatic or assisted history matching methods (AHM) aim to reduce the manual work done by reservoir engineers to obtain a consistent simulation model with reservoir performance data. In this context we can consider two different aspects. The straightforward approach is automatic generation of the reservoir simulator input files, reading the output files and visualization of the results in order to compare different simulation results with observed data. The more difficult task which is usually considered as the research area in AHM, is the process of generating a set of reservoir parameters and improving the quality of the obtained matched during the study.

We start the literature review with traditional history matching methods which required many mathematical calculations and provided a single solution at the end. We then continue the journey into the new era of history matching where advanced optimization algorithms are employed; simulations are performed in high performance computing (HPC) environments and the end results come as multiple history-matched realizations of the reservoir.

2.1.2 Old Era of History Matching

The first research on adjustment of reservoir parameters probably was done by Kruger [1960]. He pointed out that there should be an agreement between calculated and measured pressure data. Kruger proposed a numerical method for a mathematical model of a reservoir in which the areal permeability distribution was adjusted in order to match the past reservoir performance in flooding or cycling projects. This study led to a conclusion that in order to have a reliable prediction there is a definite need to validate the reservoir model by conditioning it to the production data. Jacquard and Jain [1965] demonstrated an automated history matching procedure based on variation analysis in electric networking for a two dimensional case.

Coats et al. [1968] employed least square and linear programming to determine the reservoir description from the given performance data. Their study considered single and two phase flow in three example reservoirs. They used a random search for the parameters and linear programming for bounding the solution values of the parameters. This work demonstrated the importance of parameterization in history matching. They

noticed that final history match results were better when the inverse of permeability ($1/k$) was used instead of permeability itself.

Slater and Durrer [1970] used a gradient method to minimize the differences between measured and calculated pressure. Their method was a modification of the Jacquard method which requires less computational time and finds the direction (changing error sign) and step size (10% to 15% of original parameter value) of the corrections that should be made in order to minimize the difference between observed and calculated data. Several difficulties were observed as a conclusion in this study. High sensitivity and a strong non-linear relationship between the error and low permeability values made it difficult to find the correct values by the gradient method in low permeable, tight regions. Also they concluded that it was difficult to decide when to switch from working on one parameter to another one (porosity to permeability).

Thomas [et al. 1971] used a Gauss-Newton least-square procedure for history matching and comparison of the results with Coats' work [1968] method showed that the new method got similar results from fewer simulations.

In the mid 1970s, optimal control theory became a popular method for obtaining history matched models. Chavent et al. [1973] and Chen et al. [1973] used optimal control theory for automatic history matching in single phase flows with constant compressibility. Dougherty and Kheirkhah [1975] used this technique for real-gas systems. Also in this study it was concluded that unknown parameters obtained in history matching are not unique and different starting values for parameters yield different final values.

Gavalas et al. [1976] introduced a Bayesian framework for history matching. In this method, the unknown vector of the discretized reservoir parameter was viewed as a random variable having a mean α and a prior covariance matrix C. The values for the mean and covariance matrix are obtained from geological information measured in the field. Results obtained by Shah et al. [1978] indicated that if a reliable *a priori* statistics are available, using a Bayesian approach will lead to a smaller variance of the estimation error in comparison with two parameterization methods (zonation parameterization without considering geological information and parameterization with

sensitivity vectors). Shah's paper also concluded that only a modest number of parameters can be determined in a successful history match. They state that total uncertainty reaches its minimum only at a particular level of parameterization or alternatively number of zones. This confirms the results previously obtained by Coats et al. [1968] in which they showed that history matching schemes are not able to solve problems in which there are many unknown parameters.

Pruess et al. [1980] used the SHAFT 79 simulator developed in Lawrence Berkley Laboratory for history matching. This can be considered as a pioneering work in coupling the history matching framework with numerical reservoir simulation software.

In 1986 Watson presented a history matching method based on a modification of the Gauss-Newton method [Watson and Lee, 1986]. This was followed by another work in which the procedure of generating sensitivity coefficients in the Gauss-Newton method was modified to reduce required computational efforts [Tan, 1995].

Marsily et al. [1987] introduced the use of geostatistical methods in inverse modeling. They used the concept of pilot points in which the reservoir parameters are being estimated in a limited number of points and the remaining values at other locations are obtained by kriging. Fasanino et al. [1986] applied this method in a real gas reservoir problem. Use of geostatistical methods led to a lower number of unknown parameters that should be estimated; however coupling this approach with a gradient optimization algorithm still suffered from the major drawback that only a limited number of unknown parameters that can be successfully estimated.

Zuber et al. [1987] used history matching to obtain properties of coalbed methane (CBM) reservoirs and suggested that in order to obtain reliable results, accurate field and laboratory measurements should accompany the history matching process.

Yang and Watson [1988] used a constrained optimization method introduced by Powell [1978] and a self scaling variable-metric (SSVM) [Oren and Luenberger, 1974] for the history matching of two-phase 1-D and 2-D cases. They compared the results with steepest descent and Nazareth's conjugate gradient [1977]. This comparison confirmed that Powell's method and SSVM generally performed better than steepest descent and

the conjugate gradient method, with SSVM being more efficient than Powell's algorithm in higher dimensional search spaces.

Anterion et al. [1989] presented an analytical approach for computing gradients and implemented it in a three-phase three-dimensional simulator. They used a gradient minimization technique for history matching of two reservoir simulation cases. As the authors mention in this paper, these types of methods are not intelligent enough to know which parameters should be perturbed in order to obtain a good match and leaves this task to the user. Thus there exists the need for the experience of a reservoir engineer to specify the critical parameters that should be changed.

In 1992, while researchers focused on introducing new algorithms for automatic history matching, Watkins et al. [1992] highlighted the importance of user interaction with these frameworks. In this context, the reservoir engineer's view may be changed over the course of history matching and thus we need to interact with the framework. This interaction can be performed by supplying initial estimates of parameters based on engineering grounds or it can be in the form of a dynamic interaction with the optimization procedure to determine the specific search directions or other essential work that the engineer prefers to include in the process. This idea forms a basic platform for many commercial history matching softwares. Watkins et al. [1992] also stated that the information supplied as the prior search boundaries may be subject to considerable error which brings the possibility of using stochastic methods for uncertainty quantification.

A year later Parish et al. [1993] introduced a knowledge based system (KBS) as an addition to the interaction steps between the reservoir engineer and the automatic history matching framework. In a KBS, knowledge of the reservoir engineer about a field is stored in a database in the form of simple linguistic if-then rules. Parish et al. [1993], as an example, provided the following statement:

if porosity is low *and* permeability is high *then* reservoir is probably fractured

Based on the above simple rule, a good choice for varying a parameter may be the fracture permeability. The goal of this system was to help the reservoir engineer in

interpretation of large data volumes produced in a history matching study. The KBS can also check suitability of base model as a good starting point for history matching. In this loop, the KBS suggests some modifications to the simulation case and the engineer has the opportunity to accept this suggestion or reject it based on his experience from the field. Then this decision can be recorded in the knowledge base to update the rules of the system and be used in future applications.

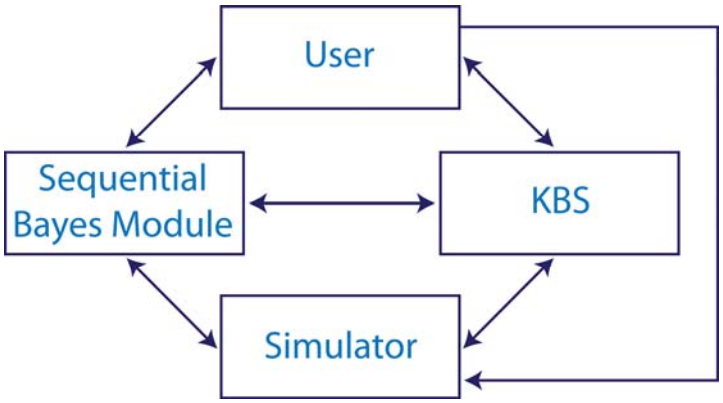

Figure 2: Framework of knowledge based system (KBS) [Parish et al. 1993]

Experimental design and response surface methods entered the reservoir engineering arena in the early 90s. Experimental design was first developed in the 1920s and 1930, by mathematician Sir Ronald A. Fisher for agricultural applications. Experimental design is a description of the different parameters settings as the inputs to the problem model. In an efficient experimental design one aims to construct design settings in order to obtain the maximum possible information from the minimum number of model evaluations. As an early attempt, Damsleth applied experimental design to a field study in the North Sea to obtain parameter sensitivities with minimum number of simulations [Damsleth et al. 1992]. In this approach, the relationship between the reservoir response and a number of input parameters was approximated with a smooth parametric function. According to Damsleth, this approach helped to reduce the number of required simulations by 30-40% in comparison with frameworks that vary one parameter at a time.

Eide et al. [1994] applied the response surface method and experimental design to the automatic history matching of a synthetic case. A response surface is a simplified

relation between the simulator input and output which can be useful in a case that computer intensive runs are involved. The response surface can be used to predict the results of simulations from other combinations of input parameters that already have been tried.

The quality of a response surface method depends on some issues like its design. It may be restricted by the number of input parameters [Eide et al. 1994] as with growing the number of system input parameters, the function evaluations required to build a high quality response surface will increase drastically. O'Dell and Lamers [2005] showed that in cases with significant dependency between parameters, the shape of response surface may not be captured by a limited number of simulations performed in the design stage of an experimental design. They pointed out that experimental design is useful at later stages of field development where a development plan is proposed and more data about the reservoir and wells are available.

In 1992, Ouenes made a breakthrough by introducing the application of a global optimization algorithm called simulated annealing for history matching which does not require calculation of gradients. They used this new method extensively for a number of cases, including characterization of a gas storage reservoir [Ouenes et al. 1992 – A], for interpreting gas/water laboratory corefloods [Ouenes et al. 1992 – B], for history matching problem in a gas reservoir [Ouenes et al. 1993 – A] and in an oil reservoir [Sultan et al. 1993]. In parallel research, Sen et al. [1995] also applied simulated annealing to a set of outcrop data to reproduce the permeability distribution. Sen suggested a heat-bath algorithm as a new version of simulated annealing which in comparison with traditional simulated annealing method, performed better in large simulation cases.

Later Ouenes proposed a parallel version of simulated annealing for reservoir characterization [Ouenes et al. 1993 – B]. He used this approach for obtaining history matched models using parallel computers [Ouenes et al. 1995]. This was the first time that the power of parallel computing was used in history matching.

By the mid 90s, assisted/automatic history matching was becoming a familiar concept for the reservoir engineering community. Many forward-looking ideas were entered into

this area during thirty years of research in the automatic history matching field. To summarize the step changes until this date, we can use figure 3.

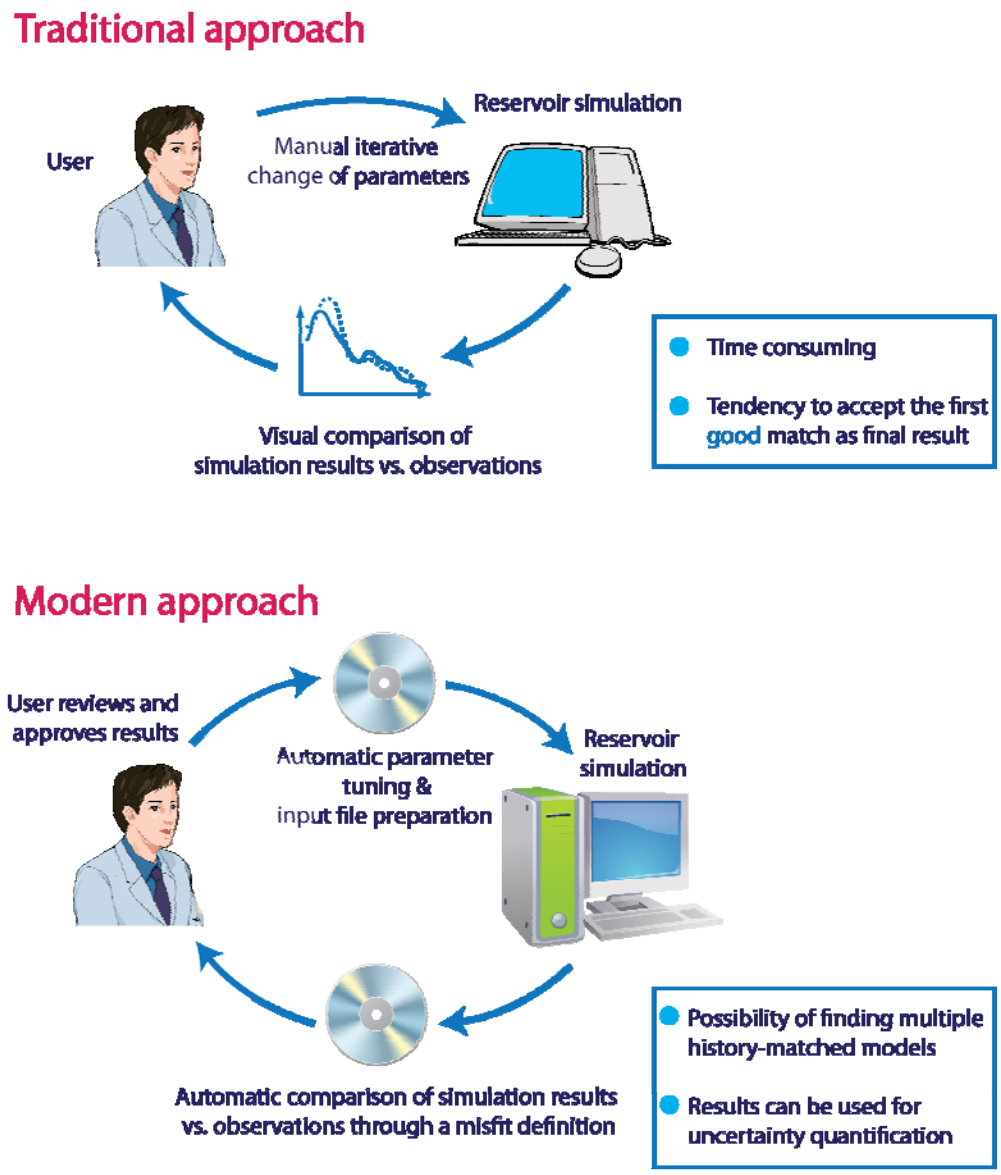

Figure 3: Traditional vs. modern approaches to history matching problem

By the end of the 90s and after several ground-breaking changes, the general framework of history matching was well established. Further research has been mainly focused on improving the workflows considering different areas like algorithms used for obtaining history-matched models or error modeling work. The modern era includes the maturity of previous methods, application of new stochastic methods and the entry of soft computing techniques into history matching domain.

In the next section, we will look at this modern era of history matching and will briefly review different improvements made in automatic history matching in this period.

2.1.3 Modern Era of History Matching

The modern era of history matching started two decades ago. At this time, the importance of multiple history matched models became widely understood and automatic history matching methods with the capability to generate multiple models were becoming an interesting area of research. It was in early 90's that the trend in history matching was geared towards generating multiple history matched models [Palatnic et al. 1993] [Tyler et al. 1993].

When stochastic methods entered the reservoir engineering arena, many works showed that simple optimization methods are not good tools for solving complex history matching problems [Bush and Carter, 1996]. Figure 4 shows the misfit function for a simple history matching problem with one unknown parameter. In this example, we will need to identify not only the global minimum, but also multiple local minima in the search space. The problem of local minima becomes particularly important in reservoir prediction studies. Finding multiple local minima are essential for realistic quantification of prediction uncertainty, which is often not the case in many traditional optimization problems. It has been shown that a single best history matched model is not necessarily a good predictor for future performance of a reservoir [Tavasolli et al. 2004]. On the other hand, it is difficult and inefficient to achieve this goal using conventional Monte Carlo approaches because these methods are not intelligent enough to maximize the number of multiple good-fitting models in a limited number of simulations.

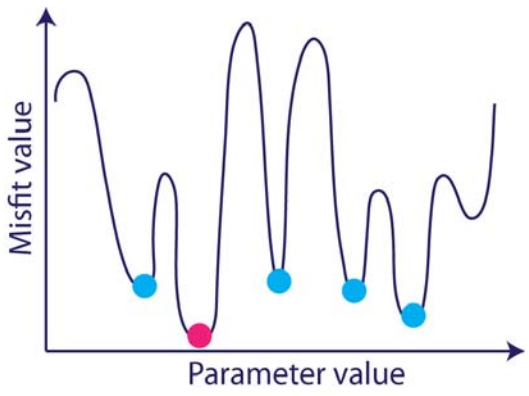

Figure 4: Global minimum (pink) and multiple local minima (blue) in history matching problem

Stochastic methods are often seen as a good choice for history matching for several reasons. We are able to extensively control their behavior. Koppen, in an interesting discussion, introduced a new viewpoint for comparison of the algorithms based on their diversity [Koppen, 2004]. By diversity of the algorithms, one means the distribution of optimization algorithms, according to their different configurations, runtime parameter settings and pseudo-random number sequences. He calls possible outcomes as *instances* of algorithms and ranks different methods based on this index. He concludes that in this framework, the random search technique is the worst, the middle rank belongs to hill climbing techniques and in the best rank we have evolutionary population-based algorithms. Population-based systems are composed of multiple intelligent individuals that utilize the interactions among members to improve the quality of solutions. Stochastic population-based optimization algorithms can give (at least theoretically) any search sequence that is possible.

Novel adaptive stochastic methods also provide the opportunity to balance exploration and exploitation while searching for optimal solutions. Exploration refers to the search of different areas in the parameter space while exploitation is the refinement of the previously visited regions to find better answers. Wan and Igusa [2003] discussed the need for greater accuracy in regions of the search space corresponding to low objective function values and the benefits of adaptive sampling methods to satisfy this purpose.

Wetter and Wright [2004] made a comparison between 9 deterministic and population-based probabilistic algorithms for the case of simulation-based optimization. Like the history matching problem, in their application, the optimization algorithm is coupled with a simulation program for building design and energy analysis. They concluded that probabilistic methods are more likely to find the optimum solutions in optimization problems where there are discontinuities in the cost function due to numerical errors of modeling.

Population-based algorithms are also more robust in comparison with point-based methods when we deal with the optimization of noisy objective functions [Nissen and Propach, 1998]. Population-based methods use a set of agents during the optimization and are thus less affected by the quality of individual solutions. In the next section we

shall review some of the stochastic algorithms introduced in the modern era of history matching.

2.1.3.1 Genetic Algorithm (GA)

Genetic algorithm which belongs to the group of evolutionary algorithms, was first proposed by John Henry Holland [1975]. The main idea of GA is based on Darwin's theory of natural evolution. It employs the survival of the fittest strategy to guide the optimization process within the search space. Shortly after the first application of GA in petroleum engineering [Goldberg, 1995], this method became one of the most popular optimization algorithms to solve petroleum engineering problems. It has been applied to oilfield development [Tupac et al. 2007], well placement optimization [Emerick et al. 2009], optimizing well starting times and schedule in the field [Jutila and Goodwin, 2006], production strategy optimization [Nogueira and Schiozer, 2009], oil recovery optimization in CO_2 flooding [Chen et al. 2009] and for investment decision making in oil and gas business [Sarich, 2001].

For history matching purposes many authors have used genetic algorithms to condition reservoir simulation models to dynamic data. One of the very first applications of GA in reservoir modeling was by Sen [et al. 1995]. They applied a genetic algorithm to a set of outcrop data and compared the results with two versions of simulated annealing. It was concluded that the performance of the genetic algorithm is highly dependent on the tuning parameters chosen and it usually requires more function evaluations than simulated annealing.

Romero et al. [2000-A] also used genetic algorithms for history matching of reservoir simulation cases. Later they proposed a modified version of genetic algorithm with different chromosomes for different types of reservoir parameters [Romero et al. 2000-B]. Testing the proposed method on a synthetic field study, they report this approach was less sensitive to algorithm parameter settings. Ballester and Carter [2007] used a modified version of genetic algorithms and parallel computing environment to obtain history matching models. They also applied a clustering method to identify the different types of models obtained after history matching.

Williams [et al. 2004] introduced a new concept in history matching and uncertainty quantification based on the slogan of “as complex as necessary”. The so called top down reservoir modeling (TDRM) approach as a semi automatic history matching tool is now a main part of BP reservoir simulation workflow. The TDRM workflow is equipped with a genetic algorithm optimization engine in order to find multiple history matched models. The TDRM approach has been widely used in many real life applications both at the appraisal stage and in mature fields and has achieved successful results [Walker and Pettigrew, 2006], [Walker et al. 2008] and [Litvak and Patrick, 2009].

Erbas [2006] also used a genetic algorithm for the history matching of different synthetic and real field reservoirs. Genetic algorithms have been also combined with non-linear proxy models to find history matched models in less time [Castellini et al. 2006].

2.1.3.2 Evolutionary Strategies (ES)

The evolutionary strategy was introduced by Ingo Rechenberg [1965] and Hans-Paul Schwefel [1968] from the Technical University of Berlin. This technique is efficient for global and local optimization of continuous and discrete search parameters. The evolutionary strategy also uses mutation and recombination operators to find optimum solutions, with mutation being emphasized over recombination.

Evolutionary strategies have been successfully applied to tackle history matching problems [Schulze-Riegert et al. 2001, Haase et al. 2006]. Selberg et al. [2007] combined evolutionary strategies with an experimental design framework for the history matching of a large gas condensate reservoir model in the North Sea. MEPO is a commercial history matching software that has benefited from an optimization engine based on evolutionary strategies [Choudhary et al. 2007].

2.1.3.3 Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) is a population-based stochastic optimization method. It was originally introduced by Kennedy and Eberhart [1995] by studying the social behavior of a flock of birds. It has been applied in a wide range of challenging

computational and engineering optimization problems. For a review of different application areas interested readers can refer to Banks et al. [2007], Eberhart and Shi [2007] and Lazinica [2009].

In reservoir engineering, Kathrada applied PSO and a hierarchical clustering algorithm to obtain history-matched models and tested this approach in a synthetic case [Kathrada 2009]. Mohamed et al. [2009] compared the efficiency of PSO for history matching with the Neighbourhood Algorithm (NA) and Hamiltonian Monte Carlo (HMC). She concluded that PSO obtains good history matched models from fewer simulations for their particular model. Fernandez also applied the PSO algorithm for two reservoir engineering applications: seismic history matching and production optimization [Fernandez et al. 2009].

2.1.3.4 Scatter Search (SS)

Scatter search is classified as a stochastic population based optimization algorithm and was first introduced by Fred Glover [1977]. Scatter search works with a set of solutions which is called the reference set and the algorithm tries to improve the quality of this reference set by making linear combination of solutions to create new ones.

Sousa et al. [2006] applied the scatter search method to the history matching of two simple reservoir cases with a small number of uncertain parameters. The scatter search method seems promising for the history matching problem based on the limited results reported by Sousa; however the efficiency of scatter search for problems with a large number of unknown parameters should be a topic of further research. In this framework, history matching is formulated as a combinatorial optimization (CO) problem which requires the uncertainty domain of the parameters to be discretized. In order to get accurate results we have to reduce the space between discretized parameter values and this increases the number of points and possible solutions. Finding optimum models for this large number of combinations of points while having a small number of objective function evaluations is not an easy task for an optimization method. A possible solution for this drawback is coupling scatter search with a proper proxy model that enables us to run a relatively large number of simulations for the discretized search space.

2.1.3.5 Simultaneous Perturbation Stochastic Approximation

Simultaneous Perturbation Stochastic Approximation (SPSA) is a global optimization algorithm which has recently attracted attention for solving petroleum engineering problems. The SPSA was introduced by James C, Spall and works by performing a stochastic simultaneous perturbation of all model parameters to generate a search direction in each iteration. Branchs et al. [2006] used the SPSA algorithm for history matching a 2D reservoir model. Gao et al. [2007] also used a modified version of the basic SPSA algorithm for history matching and improved convergence rate. Jia et al. [2009] performed history matching of steam assisted gravity drainage (SAGD) experiments using the SPSA algorithm.

2.1.3.6 Proxy Models

The goal of a proxy model is to replace the actual simulation with a model which does not require the actual forward simulation to be done by the reservoir simulator. Mohaghegh [2006] used surrogate reservoir models (SRM) and Monte Carlo simulation for uncertainty quantification. SRMs are approximate models that can mimic the behavior of full field models. This enables Monte Carlo analysis to be performed on large fields as simulation results can be obtained for each case within seconds [Mohaghegh et al. 2006]. SRM reduces the dimensionality of the problem using fuzzy pattern recognition. The idea is to identify key performance indicators (KPI) and layers which play the most important role in the performance of reservoir using fuzzy patterns. SRM should be developed to achieve specific and predetermined goals (for example the prediction of oil rate) in field studies. SRMs cannot replace conventional reservoir simulators or act as global solution. Selection of data for training and validation of the model is another challenge that should be addressed in using a SRM.

Christie et al. [2006] used neural networks as a proxy method instead of expensive forward simulation runs. Neural networks are trained by data obtained during the initial search stages of parameter space. Neural networks can learn from the information about the relationship of input parameters and corresponding misfit values. Once neural networks are trained, they can be used to predict the misfit values for other combinations of input parameters in history matching. Christie et al. [2006] concluded that in order to ensure we find good models in other regions of the search space, we may continue to run the sampling algorithm in the regions that we have missed in the

initialization step. Neural networks may predict bad fitting models in these regions of the search spaces, since these models were not present in the training stage. This conclusion suggests that the initialization of the search and the training stage of the neural networks are very important in using this proxy method.

Zubarev [2009] presented a comparative study of several proxy modeling and full numerical simulation approaches in assisted history matching frameworks. He used polynomial regression model, multivariate kriging, thin-plate splines and artificial neural networks in this study and concluded that all proxy methods strongly depend on the model complexity, dimensions of the design space and quality of the input dataset.

2.1.3.7 Ensemble Kalman Filters (EnKF)

The Ensemble Kalman Filters (EnKF) originated as a version of the Kalman Filter and was first developed by Evensen [1994]. EnKF, by nature, is a stochastic method. Like population-based algorithms, EnKF uses an ensemble of models rather than a single solution. EnKF uses piecewise assimilation of data, forward in time as a major concept which makes EnKF different from other stochastic population-based optimization algorithms in which we generally do not consider the time dimension during the optimization run. It is considered to be very promising because it is flexible for a real-time modeling and data incorporation from new measurements.

Since the first application of EnKF to reservoir engineering by Naevdal et al. [2002], this method has enjoyed a growing popularity for history matching. There are many successful applications of EnKF in reservoir engineering and history matching which are reported in the literature. Liu and Oliver [2005] used EnKF for history matching geological facies and compared the results with the gradient based randomized-maximum-likelihood (RML) method. They obtained better history matched models from fewer simulations using EnKF compared to the gradient method. More examples of EnKF being successfully applied to history matching problems can be found in Naevdal [2003] and Bianco et al. [2007].

The Ensemble Kalman Filter, like any other method, suffers from some drawbacks. Petrie [2008] investigated three problems that may occur in EnKF by undersampling. Undersampling happens when the number of ensemble members is small compared to

the size of the state. Three problems discussed in her thesis are inbreeding, filter divergence and the development of long range spurious correlations. She reviewed two recently proposed strategies to overcome these problems: covariance inflation [Anderson, 2009] and covariance localization [Hamill et al. 2009]. In covariance inflation one applies an inflation factor to forecast error covariance in order to overcome the problem of inbreeding. In covariance localization one tries to remove the long range spurious correlations. She reported a problem in implementing a method for localization called the Schur product. She confirmed that for both cases the estimation error was increased compared to the truth solution which is certainly not desirable. For her specific case study, she reported that neither of these two methods can effectively help prevent the problems of EnKF.

Jardak et al. [2009] in a recent publication compared EnKF with the particle filter (PF) and maximum likelihood ensemble filter (MLEF). They reported that for nonlinear observation operators, standard EnKF, even when applying localization and inflation does not provide good estimates. Valles and Naevdal [2008] investigated the use of paired and coupled EnKF as a possible remedy to the inconsistency problem in EnKF which was observed by Lorentzen et al. [2005]. Valles and Naevdal [2008] proposed alternative approaches including a paired EnKF where the gain matrix of two sub-ensembles was used to update another sub-ensemble and coupled EnKF where one Kalman gain matrix was used to update both sub-ensembles. They reported that neither of these methods could provide satisfactory results in a highly non linear problem. Also in their approach only half of the initial ensemble is of interest which can be seen as a downside for the method.

Franssen and Kinzelbach [2009] compared EnKF and sequential self-calibration (SSC) methods for solving groundwater problems and conclude that the performance of EnKF can be worse compared to SSC in the case where the problem exhibits strongly non-multi Gaussian properties. They also point out another issue with EnKF where the numbers of observations or measurements play an important role in the CPU time used by EnKF. Inversion of the resulting matrix $n \times n$ (where n is the number of observations) where a large number of observations exist, may require a huge amount of CPU time, comparing to the forward simulations needed.

In the petroleum engineering context, it has been stated in the literature [Sarma and Chen, 2009] that standard EnKF is suitable only for systems characterized by two-point geostatistics. Although standard EnKF can provide a good match for observed data, its prediction capability in complex non-Gaussian geological cases such as channelized systems is questionable. They proposed a kernel method to create a nonlinear generalization of EnKF that is able to handle non-Gaussian random fields.

Another concern in the EnKF framework is the computational time required. Although approaches like coupling EnKF with polynomial chaos [Saad and Ghanem, 2009] have been proposed to reduce computational cost, this topic is still an open research area. Another issue that is generally considered as a source of concern in using EnKF for history matching is its ability to keep geological structure. There exist some approaches for preserving geological structures. For example Lawniczak et al. [2008] used a Ensemble Multiscale Filter (EnMSF) for history matching of a 2-D simple simulation case. EnMSF is believed to be more powerful in preserving the geological structure.

2.1.3.8 Hybrid of EnKF with Stochastic Algorithms

Pajonk [et al. 2008] combined EnKF and evolutionary strategies and proposed a hybrid method of EnKF-ES in which a simple ES is used to improve the estimation of a static model parameter between forecast steps of the model during an EnKF run. They tested the new method for optimization of a highly non-linear model and concluded that the new approach could improve parameter estimation. Schulze-Riegert [et al. 2009] presented the application of this workflow to a large scale history matching problem – the Brugge reservoir model.

Figure 5 summarizes the different approaches developed during 50 years of research on assisted history matching from 1960 to 2010.

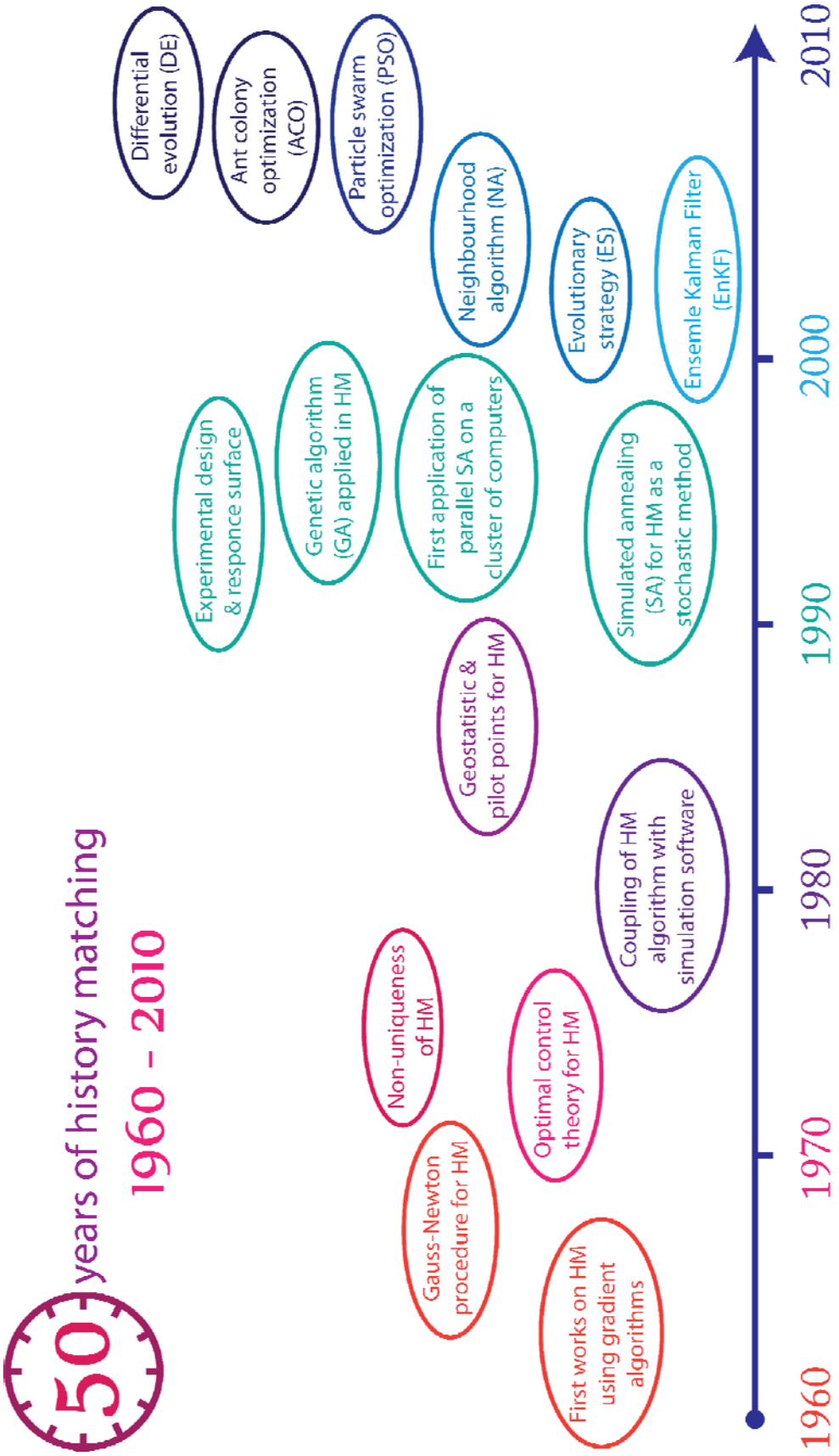

Figure 5: 50 years of history matching

2.1.4 History Matching Methods: Rises and Falls

Davies [2010] recently presented a very interesting comparison between different methods used for automatic history matching. The statistics came from searching Google scholar and although it is not guaranteed to find every single paper, it can be a good approximation for the trends. Figure 6 presents the number of papers on "history matching" and "automatic history matching" during 1960-2009.

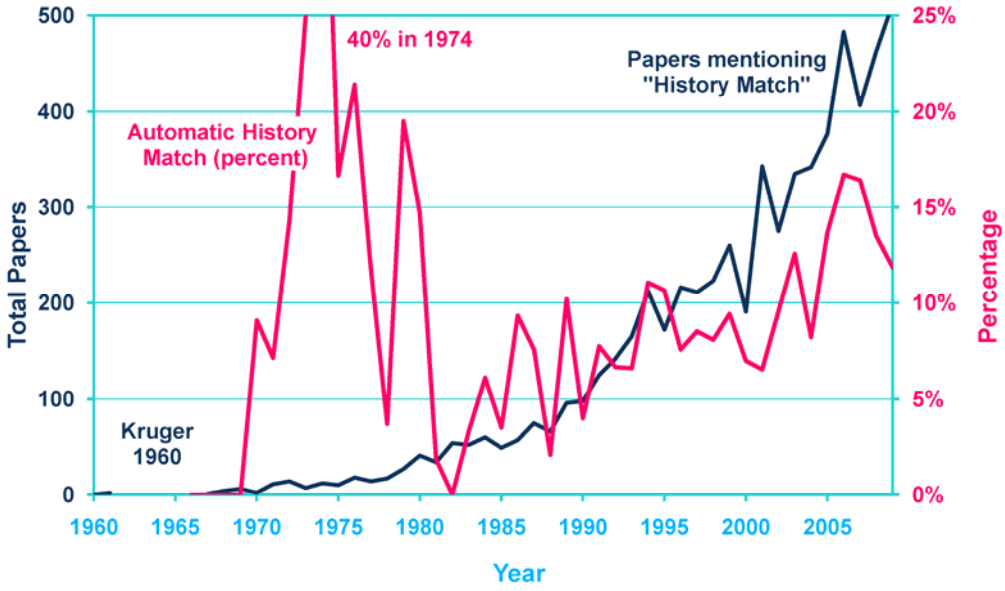

Figure 6: Trends of history matching and automatic history matching [Davies, 2010]

This graph shows a constant increase in the number of papers on history matching published per year during the past 50 years and thus the fact that the industry has accepted the necessity and importance of history matching in reservoir engineering workflows. Looking at the trend of automatic history matching, we see it had a huge popularity immediately after introducing the concept. In the mid 70s, almost 40% of the papers on history matching were devoted to the principle and methods of automating the process. However by the early 80s automatic history matching had lost its popularity, probably due to the fact that initial algorithms could not satisfy the needs of industry. The trend again started to rise after the introduction of novel algorithms in the late 80s and early 90s. The number of papers on automatic history matching has been constantly increasing during recent years due to the success of the proposed methodologies.

We can also look at some well-known methods used in automatic history matching. Figure 7 presents the percentage of papers in the automatic history matching field using different approaches.

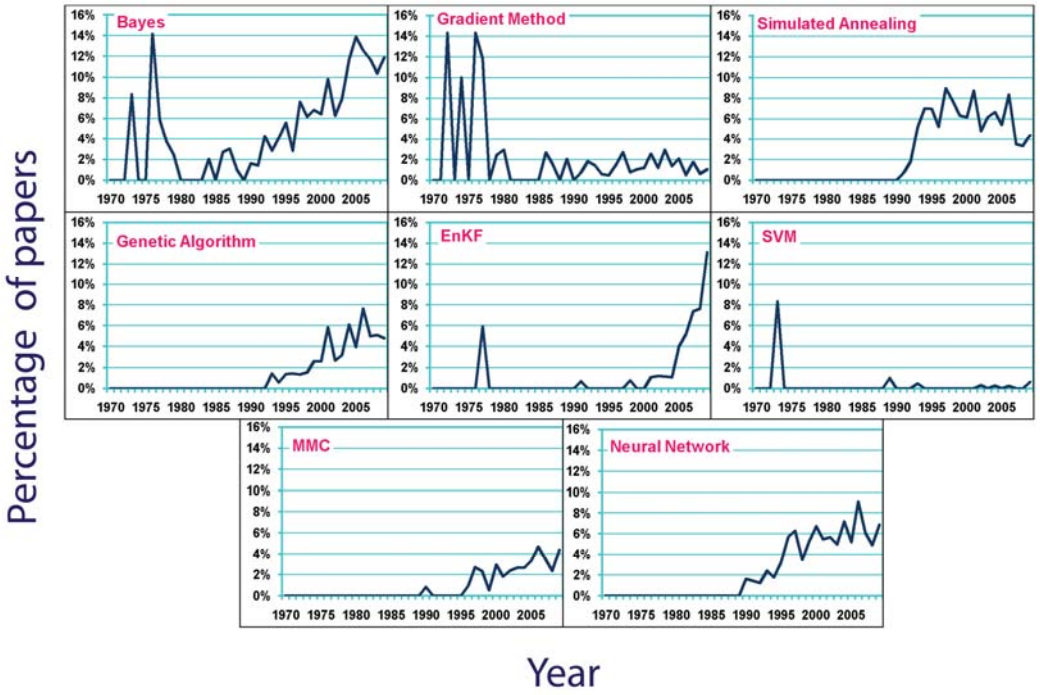

Figure 7: Trends of well-known methods for automatic history matching [Davies, 2010]

Examining this figure reveals some interesting points. Gradient based methods were very popular during the first years of research on automatic history matching algorithms. After the introduction of stochastic methods, there has been less focus on gradient-based algorithms and techniques such as simulated annealing or genetic algorithms started to become popular. In figure 7 we can also see the constant growth in the number of papers using a Bayesian approach to history matching. Another fact in this figure is the rapid increase of popularity for the Ensemble Kalman Filter technique in recent years. Providing multiple history-matched reservoir models can be considered as the reason for the popularity of modern methods including stochastic population-based algorithms.

2.1.5 What We Do with History Matched Models?

The industry is moving towards using multiple history matched models for uncertainty quantification. We have discussed many approaches in this section for generating multiple history matched modes. Due to the non-uniqueness of the history matching

problem and the risk arising from such non-uniqueness [Yamada, 2000], we have focused on generating multiple history matched models. These models can be used to provide a range of forecast options for reservoir behavior in the future and we can then quantity this uncertainty.

In the next section, we will describe the uncertainties involved in petroleum engineering and how we can use multiple history matched models to quantify the uncertainty of production from a given reservoir.

2.2 Uncertainty of Petroleum Reservoir Models

Knowing the nature of a phenomenon is the most important step in finding ways to deal with it. It is vital to capture inherent uncertainty in order to handle the risks of a development project. Before we talk about how we quantify uncertainty of predictions, we should know about the uncertainty itself, its difference with error, the basic types of uncertainty and the areas in which it can affect the oil and gas industry.

2.2.1 What is Uncertainty?

The answer to this question is not the same for different groups of people. Each group will focus on a different usage of the term ‘uncertainty’, depending on their areas of interest. Webster dictionary says: “uncertainty may range from a falling short of certainty to an almost complete lack of conviction or knowledge, especially about an outcome or result”. The American Institute for Aeronautics and Astronautics (AIAA) guidelines define this term as: “A potential deficiency in any phase or activity of the modeling process that is due to lack of knowledge” [AIAA, 1998]. The International Organization for Standardization (ISO) describes uncertainty as: “A parameter (this may be for example a standard deviation or the width of a confidence interval) associated with the result of a measurement that characterizes the dispersion of the values that could reasonably be attributed to the measured” [ISO, 1993].

The words of "uncertainty" and "error" are usually used interchangeable in common everyday language and their concepts have been confused with each other for a long time. Error can be defined as a recognizable deficiency in any phase or activity of

modeling and simulation that is not due to lack of knowledge [AIAA, 1998]. As it is obvious from the above definition, the key difference between uncertainty and error is lack of knowledge about the processes that we are going to model. The key word 'potential' in the definition of uncertainty indicates that deficiencies may or may not exist. An uncertainty in most cases does not have a sign.

The value for uncertainty should provide us with some information about how "wrong" a given value is. As most of the components that are the main source of uncertainty are unknown and many of them change during time, this value can provide a guess about the probable error.

2.2.2 Why Care about Uncertainty?

Estimates of current reserves and future hydrocarbon production determine directly the profitability of every field development project. The goodness of our reservoir recovery predictions and their uncertainty has been always considered as a major concern [Mannon, 1964], [Walstrom et al. 1967], [Dougherty and Kheirkhah, 1975]. In today's competitive market, making correct investment decisions in field development operations requires an accurate estimate of uncertainty in predictions. These uncertainty estimates reflect our confidence about the future performance of the reservoir.

The design of all wells and surface production facilities depends on the production from the reservoir. A too-high estimate will result in an over-capacity production system, while underestimating uncertainty may result in a limited production, because wells and surface facilities cannot simply handle the produced oil. Reservoir uncertainty affects the design of all elements of production systems. For example Birchenko et al. [2008] studied the impact of reservoir uncertainty on the selection of advanced well completion type.

With uncertainty being tied to every step of the field management workflow, from data acquisition to modeling and model verification, great care must be taken to understand and quantify these uncertainties in order to minimize the operational risks and increase the chance of having a successful project. Training petroleum engineers to understand and manage uncertainty is an important step in this process [Bratvold and Begg, 2006].

2.2.3 Three Types of Uncertainty

According to Blockley and Godfrey [2000], there are three types of uncertainty that we have to recognize and distinguish between them: randomness, fuzziness and incompleteness.

2.2.3.1 Randomness

Randomness uncertainty is defined as a lack of specific pattern in the variables. In the reservoir engineering context, we should be aware of possible randomness of reservoir properties as a source of uncertainty, especially for complex geological areas where a clear pattern may not be visible. The term visible is used here because these patterns may be present, but stay hidden underground due to our lack of data. This lack of data prevents us from obtaining patterns with acceptable statistical certainty.

2.2.3.2 Fuzziness

Fuzziness is defined as an imprecision of definition. This imprecision might be due to the measurement process or the way we decide to express the parameters of interest. Fuzziness tends to challenge our concern of precision. Zadeh has a famous sentence which says “As complexity rises, precise statements lose meaning and meaningful statements lose precision”. The human reasoning and decision making process is not crisp.

This type of uncertainty can be treated with fuzzy logic. Fuzzy logic can accomplish the task of uncertainty representation through fuzzy set theory. Fuzzy set theory was introduced by Zadeh [1965]. Unlike the Aristotelian logic which looks at the world in a bivalent crisp manner like black and white, fuzzy logic looks at the transition zones between numbers or grey colors. In fuzzy sets everything is a matter of degree and every object belongs to a set to a certain degree. This theory was developed to obtain approximate solutions in problems with vague description. Figure 8 presents a fuzzy description of a rock sample and its comparison with an exact description.

Figure 8: Exact vs. fuzzy description of a reservoir rock sample

We demonstrate the smooth transition between sets in fuzzy logic with the following example. Imagine we are going to determine a permeability cut-off for net to gross ratio determination. A permeability of 45 mD is determined as the cut-off value. Given a permeability value of 44 mD, a normal approach would classify this value as a non-pay zone while there is only a small difference between this value and the determined cut-off permeability. What about a permeability of 46 mD?

The crisp set theory only allows us to assign the permeability value of 44 into a non-pay zone set, while the values of permeability obtained from well logs are vague. This is a clear example where fuzzy sets could be helpful. Fuzzy set A is defined by a real value function $\mu_A(x)$ = [0,1] called the membership function of A, which assigns to every element of x a real number between 0 and 1 (degree of membership). Figure 9 illustrates the concept of fuzzy membership for the above example.

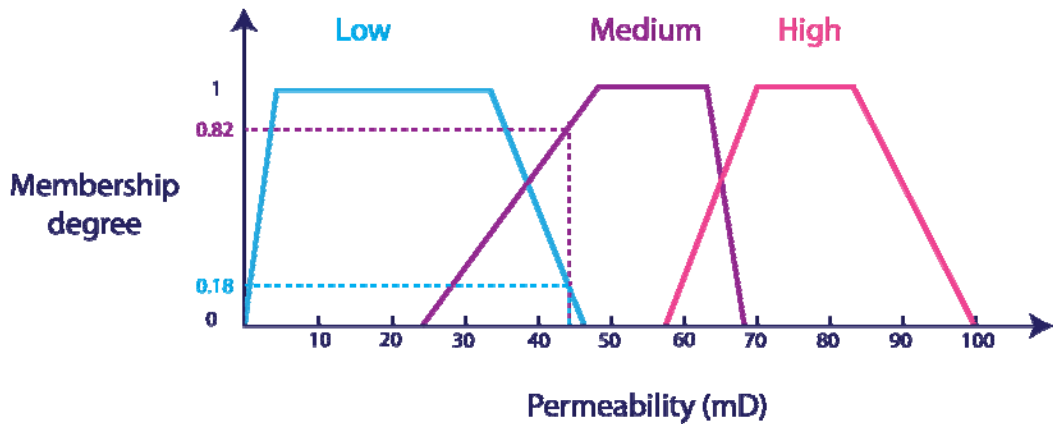

Figure 9: Fuzzy membership functions for estimation of permeability

Based on the above definition for fuzzy membership, when permeability is 44 mD, it is considered to have a low permeability with membership degree of 0.18 and a medium permeability with a degree of 0.82 which can be expressed as:

μ_{Low} (44 mD) = 0.18 and μ_{Medium} (44 mD) = 0.82

In this way we can have a better permeability characterization. Fuzzy variables and parameter estimation have been shown to be helpful in uncertainty quantification [Moller et al. 2002].

2.3.2.1 Approximate Reasoning

In the fuzzy logic world, decisions can be based on fuzzy linguistic variables (high, low) and fuzzy operators (and/or). Approximate reasoning is based on fuzzy propositions of the various types in the format of simple if-then rules. To illustrate the essence of this concept, we return to our previous example and now we consider two other properties to determine the oil amount we can produce from a specific formation.

IF permeability is low AND porosity is high AND viscosity is high, THEN oil production is low.
IF permeability is high AND porosity is medium AND viscosity is medium, THEN oil production is medium.
IF permeability is high AND porosity is medium AND viscosity is low, THEN oil production is high.

Here low, medium and high labels are defined as fuzzy sets for porosity, permeability, viscosity and oil production. There are many examples where fuzzy logic has been used successfully to handle uncertainty in geoscience applications. For a review of these applications one can refer to Nikravesh et al. [2002], Wong et al. [2002] and Mohaghegh [2000].

2.2.3.3 Incompleteness

Incompleteness uncertainty can be defined as a lack of data and refers to what we do not know about the system under study and cannot be modeled. Incompleteness is the most

common source of uncertainty in petroleum engineering. There may be lack of data for many reasons. For example reservoir properties between two sampling location at two wells are uncertain. Problems in drilling or specific well completion can reduce the number of pressure measurements [Macary et al. 2005]. This lack of pressure measurements will then bring additional uncertainty to reservoir engineering studies, for example at the history matching stage. Another example is the early exploration stage where we have lack of information about the reservoir system and geological interpretation [Caumon et al. 2004]. Reserve estimates are uncertain and may change during the early life of a reservoir. Studies on more than 30 fields indicated that most reserve estimation changes are tending to occur within the first 4 years of production [Dromgoole and Speers, 1997]. This can be related to a lack of information in early stages of reservoir life.

Uncertainty due to incompleteness can be divided into two sub-sections:

A) Incompleteness that we do know that we do not know

This part can be considered in two different ways. The first way is that there are certain types of information that we know are available, but we are not using them or we do not want to acquire them, simply because it is expensive. We know if we obtain this information and add them to the model, we get a better tool for predicting the system's behavior. The second way is the type of information that we know current technologies cannot capture. For example, the depth of investigation for current logging tools is only a few meters around the wellbore; so we have no information about the values of parameters we wish to know beyond this limited radius. This type of uncertainty can be reduced when more advanced technologies come to market.

B) Incompleteness that which we don't know that we do not know

This part is often the main source of failure in models when used to predict reservoir behavior. There are some facts about our reservoir that we still do not know about. Some of these unknowns may become known later in the life of reservoir and help us to improve our model and also increase production from the field. Williams et al. [2006] presented a good example of how gaining new knowledge about the phenomena that were previously unknown can help to bring life to a field that was considered to be abandoned by 2008 and thus extended production until 2030.

In statistical problems which deal with chance and probability, the first step to solve the problem is to understand and determine the basic process which generates the outcome. Once we identify this process, we can develop a mathematical representation of the process with some equations which can be used to calculate the probabilities of different outcomes. In other industries, it is easy to achieve inexpensive statistical sample data prior to major decision making process (opinion polls, market surveys, etc). However this is not the case in reservoir engineering and a new data set may come at the expense of several hundred thousand dollars or it may not be possible to obtain additional data.

2.2.4 Uncertainty Sources in Petroleum Engineering

The world of petroleum science is not unaware of uncertainty. Almost 35 years ago Capen indicated that the uncertainty is not estimated well. He reported some examples that clearly showed there is a universal tendency to underestimate uncertainty [Capen, 1976].

Garb [1988] identified three types of uncertainty in the petroleum industry: technical, economical and political. Technical uncertainties are the largest focus area in a field development study. Technical uncertainties start at the appraisal stage and continue until the last barrel of oil is produced. For example, original hydrocarbon in place (OHIP) estimation is one of the most important and crucial values to be determined in early reservoir development. It depends on the volume proportion of the reservoir that contribute to production or the net to gross (NTG) value. This ratio is computed by determining appropriate cut-offs for the layers. Any uncertainty in NTG determination may greatly impact the OHIP estimation [Sharma et al. 2008, Journel and Bitanov, 2004]. Other examples of technical uncertainties are reservoir structural uncertainties such as top horizon positioning, gross thickness and fault positing. Top horizon uncertainty is due to errors in picking the horizon from seismic and time-to-depth conversion errors [Gazet et al. 2009]. Gross rock volume is controlled mainly by the structure of the horizons and faults. Horizon correlation across faults can be a major source of uncertainty even if markers are available on both sides of the fault. For wells located near a fault, a small change in the fault position can strongly affect the reservoir production. Mature field forecasts are usually believed to be more accurate; however these fields are also not immune from uncertainty. There are several examples

describing the uncertainty in mature fields with significant amounts of data [Williams and Lond, 2006, Friedel et al. 2009].

Economical uncertainties are due to lack of knowledge about oil prices, drilling operations costs, prices of equipment, etc. Past experience has shown that oil and gas prices are at least as important as the technical and reservoir uncertainty [McMichael, 1999]. Different price models exist for estimating the oil price based on historical data [Olsen, et al. 2005]. Begg and Smit [2007] presented a sensitivity analysis of price models due to economic metrics in the value of the model parameters. They concluded that the choice of model-type has a significant impact on flexibility to manage uncertainty.

Political uncertainties such as violence, strikes, boycotts, refusal to respect contracts, terrorist acts, etc, are the third factor that affect oil and gas projects. Political uncertainties are linked to economy and productivity growth [Darby et al. 2004]. There are some models developed for quantifying political uncertainties. For example Clark and Tunaru [2003] proposed a Bayesian framework to quantify political risks and its impact on foreign direct investment. However, political uncertainty is usually ignored in the petroleum industry because it is difficult to quantify or, at best, it is inserted using a simple ad-hoc adjustment to cash flow or to the discount rate. The drawback of these approaches is that they do not consider the random nature of many political uncertainties.

Frizzell [et al. 2007] addressed the importance of considering both subsurface and surface operations in uncertainty quantification. In a study performed to model the uncertainties of a gas delivery project, the authors conclude that uncertainty quantification should be done in an integrated framework [Frizzell et al. 2007]. Not only should we consider subsurface uncertainty, but we should also study the uncertainty related to pipelines, surface facilities etc. This integration of surface and subsurface disciplines is very critical in understanding the uncertainty in projects. For a successful uncertainty quantification study and correct decision making one should follow this integrated approach.

The sources of uncertainty in petroleum engineering studies can be classified into the following main groups: uncertainty in data which are mainly due to measurement inaccuracy and uncertainty in physics of the problem and computational approximation used for modeling the phenomena. In the next sections, we will examine these uncertainty sources in more detail with some examples in each area.

2.2.4.1 Inaccuracy in Measurements

There are two types of measurements used for data collection – direct and indirect. In both cases, issues with instruments and human error in recording and processing the results can be a major source of uncertainty.

In direct measurement, we can directly access the object in order to measure desired properties. For example, we obtain a core sample from our well and take it to a laboratory to measure petrophysical properties. Core samples are necessarily disturbed when extracted and initial and boundary conditions applied to measure the desired parameters may be different from corresponding reservoir conditions, or even unsuitable to reproduce them. Elkins [1972] discussed uncertainty in initial oil in place determination in unconsolidated sands resulting from porosity estimation from core samples which were altered by drilling and core sampling operations. Uncertainty in permeability determination due to coring disturbance has been addressed in Chappell and Lancaster [2007]. These examples indicate how error in sampling and measurement due to alteration while drilling may result in uncertainty. Also laboratory studies performed in different labs do not provide the same results, even if they are supplied with exact information on the procedure to be used. McPhee and Arthur [1994] published the results of a study where five laboratories were asked to determine residual oil values for a core sample and final results varied by 20%. Even then there might be uncertainty tied with the location of obtained information. Dashevskiy et al. [2006] addressed some sources of uncertainty in depth measurement in drilling operations. This can, for example, impact the interpretation of the data and origin of samples in the reservoir. We can also directly measure dynamic reservoir data such as fluid production rates or well bottomhole pressures. These measurements are also subject to uncertainty due to device or reading errors, gauge placement, thermal and wellbore effects, etc [Izgec et al. 2007, Iwegbu et al. 2007].

In indirect measurements, we do not have direct access to the area under study and we usually obtain desired values by analyzing data from well tests, logging or seismic studies. Horne [1994] discussed the uncertainties in well testing and concluded that noise in data or measurement errors may have a significant effect on the results. Azi et al. [2008] investigated these errors in pressure and rate measurements and concluded that parameters obtained from well tests should be reported with a confidence interval rather than a unique number. Another example of indirect measurements comes from seismic studies where poor data may result in a significantly different interpretation of a single seismic image regarding the structure of the reservoir. Poor seismic quality may come from data acquisition issues, velocity anomalies, presence of gas cloud, and navigation errors. For example uncertainty in amplitude variation with offset (AVO) data due to noise has been studied by Downton et al. [2007].

2.2.4.2 Errors in Simulations

In reservoir engineering studies, the quality of the simulation model is interpreted as adequacy. Adequacy is defined as there being sufficient correspondence with reality. Quality of solution refers to the ability of that method to provide the results with sufficient precision. The results of any reservoir simulation are strongly influenced by the underlying geological model. Studies have confirmed that there is no simple relationship between geological variables and subsurface flow [Milliken et al. 2008]. On the other hand, in iterative procedures used in reservoir engineering calculations there are two issues that should be considered; numerical stability of the method and its convergence speed which measures how fast a method can come to the desired solution. Christie et al. [2005] specified three main categories of simulation errors: inaccurate input data, inaccurate physics models and limited accuracy of the solutions.

Input errors are introduced into simulations by entering inaccurate petrophysical, fluid properties and other reservoir parameters into the model. For example viscosity may vary across faults, stratigraphic compartments and also with depth [Sahni, 2003]. While we don't have access to adequate samples to represent this variation, putting a single viscosity value as an input to a reservoir simulation model can be problematic. Calrson [2003] investigated some of the issues that must be considered when preparing an input deck for reservoir simulation.

Physics errors refer to our inability to capture the underlying principles of the physical system in the computer simulation model. Saleri et al. [1992] discussed the chaotic behavior of fluid flow in reservoirs. Modeling this behavior has always been a challenge and many works have tried to introduce better and more accurate flow models. For example, coupling dynamic geo-mechanical and thermo-elastic stress models with flow in porous media has yielded better physical modeling [Bachman et al. 2003, Dean et al. 2003, Zhai et al. 2009].

Many computer simulations are based on empirical correlations. These correlations are usually developed for a limited data set and their applicability for general cases are always questionable. Caldwell and Heather [2001] show an example where different equations for the calculation of water saturation provide different answers, with the difference being larger in lower porosity rocks. Clearly this could affect the volumetric calculations for reserve estimation.

Solution errors arise from the numerical equations used for modeling physics of the problem (flow simulation). These errors represent the difference between exact and approximate solutions of the flow equations. Without correct solutions of a perfect physics model with accurate input data, we do not have much chances of obtaining meaningful answers to our problem. Truncation errors, model simplifications and approximations made in solving governing flow equations contribute to solution errors. For example, upscaling and discretization errors in reservoir simulation have been reviewed in Sablok and Aziz [2008] and simulation error models for improved reservoir predictions are discussed in O'Sullivan and Christie [2006].

2.2.5 Does God Play Dice with Reservoirs?

In 1927, the physicist Werner Heisenberg published his uncertainty principle. He states that if you consider the quantum particle, there is only a *probability* about the position and momentum of that small particle in the quantum space. Niels Bohr increased the uncertainty in the same year with his *complementarily principle*. He stated that the same particle can both act as particle or wave. These two principles made Albert Einstein upset and resulted in his famous speech "God does not play dice".

In quantum mechanics, the possible states of a system are described by a "state vector". We bring the Schrodinger's cat paradox. This thought experiment was designed by Erwin Schrodinger in 1935 and presents a cat sealed in a box. The cat might be dead or alive, depending on an earlier random event. The question is: What is the state vector of the system before opening the box and making an observation? Quantum mechanics considers the state vector of the system to be 50% alive cat and 50% dead cat until the box has been opened. After making the observation, the state vector collapses on one of the above states. Similarly, any oil field and the operations, for example drilling, have state vectors which include all possible outcomes. Our ability to determine the state vector depends on our level of knowledge about the system. Extra information alters the state vector or even collapses it to a single outcome.

Most of the uncertainties we deal in petroleum engineering are epistemic. We do not have multiple random reservoirs leaping into existence in the subsurface but we have a single reservoir with properties that are known to limited accuracy at a limited number of points.

2.2.6 Approaches to Uncertainty Quantification of Reservoir Predictions

Having discovered the different sources of uncertainty in petroleum engineering, we shall now think about the ways to quantify it. Over the past few years there has been growing recognition of the need for more rigorous statistical approaches for quantifying the uncertainty in reservoir predictions. In a response to this recognition, recent years have seen a tremendous increase in the number of available methods to derive meaningful and intuitive uncertainty estimates in physical model predictions. Quantification of predictive uncertainty in reservoir forecasts provide more information for the decision making process.

Currently two different viewpoints are being considered for uncertainty quantification [Floris and Peersmann, 2002]. The first one is called the probabilistic approach which is based upon the description of stochastic models. The stochastic models are built around a single concept of the earth model. For example the geological uncertainty is represented by having a sand channel model and stochastic variation of channel parameters such as width and thickness. The second approach is scenario-based which considers several conceptually different models. For example, these different models

may come from geophysicists using different velocity models for depth conversion, thus having several top structures. It is also possible that geologists suggest several interpretations of sedimentary environments. For example, dimensions and orientation of the geological structures such as mean direction and degree of variation of palaeoflow* can be uncertain [Martinius and Naess, 2005]. For each of these different scenarios, an earth model is considered deterministically to give a single production forecast. Floris and Peersmann [2002] suggested integrating probabilistic and scenario-based approaches to have a better understanding of the uncertainties associated with decision making.

Monte Carlo techniques have been used for more than four decades in petroleum engineering for different purposes such as reserve estimation. In Monte Carlo methods, a dependent variable is defined as function of independent variables. By running large numbers of simulations and post-processing the results, the probability density function (PDF) of the dependent variable is obtained [Behrenbruch et al. 1985]. Monte Carlo methods can be very computationally intensive. As one of the first examples, Walstrom et al. [1967] used Monte Carlo simulation to evaluate uncertainty in calculation of water saturation from well logs, determination of recovery factor from material balance and obtaining recoverable oil using volumetric equation. Gen Van Horn [1970] applied Monte Carlo simulation to analyze uncertainty in estimating gas reserves.

Regionalized (generalized) sensitivity analysis (RSA) was introduced by Hornberger and Spear [1981] as a simple Monte Carlo sampling approach. In RSA method, first the plausible ranges of key model response variables are defined as the "behavior" and outside of these ranges are identified as "not the behavior". Then by sampling (often uniform) of the model parameters, values of the response parameters are computed. If the "set of parameters" results in a prediction in the "behavior" range, it is called "behavior generating". The parameter sets that do not result in the desired range are termed "non-behavior generating". Hornberger and Spear suggested comparing the cumulative distribution function (CDF) of each parameter distribution from behavior and non-behavior generating methods. If the CDF of these two classes has a significant

* Paleoflow or paleocurrent direction is the direction of flow at the time the rocks were deposited as sediments

difference for a particular parameter, then prediction of the key response variables is sensitive to that parameter.

Based on RSA approach, the Generalized Likelihood Uncertainty Estimation (GLUE) was developed by Beven and Binley [1992]. The GLUE approach replaces the binary acceptance/rejection mechanism of RSA with a "likelihood measure" [Page et al. 2004]. This measure, which assigns different levels of confidence to different parameter sets, can have a broad range starting from simple mean square error to advanced fuzzy-based definitions. Parameter sets may be sampled from any probability distribution (mostly uniform). Many works have criticized the GLUE method for not being formally Bayesian and not implementing a statistically consistent error model [Blasone et al. 2008]. Kuczera and Parenet [1998] stated that GLUE is very computationally demanding in high dimensional problems. For a detailed discussion on the deficiencies of the GLUE approach, interested readers can refer to Stedinger et al. [2008].

The multiple realization approach has also been widely used for uncertainty estimation in hydrocarbon fields [Twartz et al. 1998, van Elk et al. 2000]. In the multiple realization tree method, first a sensitivity study is performed and combined with expert opinion to determine the key uncertain variables and then a tree is formed. Probabilities are assigned to the branches of the multiple realization tree and a simulation is performed for each branch. After finishing the reservoir simulations and based on the probability of branches of the tree, a reserve distribution can be obtained.

The Efron nonparametric bootstrap [Efron and Tibshirani, 1993] is another method for exploring model uncertainty and can be described with the following steps. First we randomly sample data n times with replacement. Then the statistics of interest are computed with the new re-sampled data and the steps are repeated X times. Next, the standard deviation of the X values of the statistic is obtained and used as the measure of the distribution of statistics in the original data. It has been shown that the estimates of bootstrap methods can be significantly biased [Meyer and Booker, 2001] and different solutions have been provided to correct this bias [Steck and Jaakkola, 2004].

Markov chain Monte Carlo (McMC) is another method for estimating the probabilities. Markov Chain is a sequence of random variables $X(0)$, $X(1)$, .. $X(n)$ where the

probability distribution for $X(n)$ is determined by probability distribution of $X(n\text{-}1)$ [Cunha et al. 1998]. The set of all possible values for a particular random variable $X(i)$ (i=0,1,2,n) is called the state space. The transition probability, P_{ij}^{n} gives the probability of obtaining state j at the n^{th} location in the sequence if the random variable is in the state i at the n-1 location in the sequence. Many authors have used the original McMC technique or its extensions or hybrids for uncertainty quantification purposes in reservoir engineering [Bonet-Cunha et al. 1998, Ma et al. 2009, Emerick and Reynolds, 2010].

Other methods of uncertainty quantification include interval calculus [Moore, 1966], fuzzy logic, possibility theory and clouds formalism. Fuzzy logic has been extensively used in different fields, including petroleum engineering, for uncertainty quantification [Ross, 2004, Mohaghegh, 2000]. Klir [2006] discussed the fuzzification of uncertainty theories and different methodological issues in this area. The possibility approach is also one of the available theories to represent uncertainty in dealing with imprecise and scarce knowledge. Zadeh [1978] was one of the first scientists to speak about possibility theory. The theory has been further developed by Dubois and Prade [1982] and Dubios et al. [1993]. Mauris [2008] explores the links between possibility theory and confidence intervals in cases of information shortage (very few measurements to know the underlying probability distribution). He concludes that possibility theory can be used to derive uncertainty estimates in situations where very few measurements (one or two) are available. The clouds approach is a recent development for uncertainty quantification in higher dimensional problems [Fuchs and Neumaier, 2008]. A cloud for a random variable is similar to the interval for a number. Clouds provide a concept for imprecise probability that can be used to derive quantitative conclusions.

Erbas [2006] classifies the existing methods to characterize posteriori uncertainty into three main groups. First, methods that work with the single best model with lowest misfit value (maximum likelihood model) to determine the posterior probability distribution of interest. Linearization about the maximum posteriori (LMAP) [Oliver, 1996] is an example in this category. The second group of methods use the subset of history matched models. For example, the randomized maximum likelihood (RML) method [Oliver et al. 1996] is a two-step process based on joint sampling of model and data variables and calibration of model variables to sampled data variables. The third

group of techniques consider the whole ensemble of models. Markov chain Monte Carlo (McMC) belongs to this group of methods. Oliver and Chen [2010] discuss the different aspects of various uncertainty quantification frameworks with their pros and cons. In the next paragraphs, the Bayesian framework for uncertainty quantification is described in detail.

2.2.7 Bayesian Uncertainty Quantification

The Bayesian framework for statistical inference is a systematic way to update current knowledge of a system after obtaining new information. In a Bayesian inference, the Bayes theorem is used to condition inferences about the value of some parameter of interest on the observed data. Bayes theory relates the posterior probability distribution function to a prior probability distribution and likelihood function. Bayes theorem provides a formal way to update our beliefs about probabilities when we are provided with additional information. In other words, the Bayesian approach uses all available information in order to reduce the amount of uncertainty.

Many works have demonstrated the efficiency of the Bayesian framework for uncertainty quantification. Alfaro et al. [2003] compared non-parametric bootstrapping method with the Bayesian approach and concluded that for a given data set, the Bayesian approach, on average, attached high confidence to a greater number of correct intermodes than does non-parametric bootstrapping. Douady et al. [2003] also compared Bayesian and maximum likelihood bootstrap approaches and showed the efficiency of the Bayesian approach for posterior inference.

2.2.7.1 Bayes Theorem

Bayes theorem, named after Thomas Bayes, states that the conditional or posterior probability $P(A|B)$ (probability that the event B will happen given that the event A has happened), is the ratio of the probability of the intersection of these events with respect to probability of event A. This can be expressed with equation 1:

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)} \tag{1}$$

Since the intersection of the event *A* and *B* is same as the intersection of the events *B* and *A*, then:

$$P(A \cap B) = P(A \mid B)P(B) = P(B \mid A)P(A) \quad (2)$$

Replacing equation 2 in equation 1, we have:

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)} \quad (3)$$

In equation 3, $P(A)$ is the prior or marginal probability of event *A*. It is called the prior probability because the information about event *B* is not considered; $P(B|A)$ is called likelihood or conditional probability of event *B* given event *A*, and $P(B)$ is the prior or marginal probability of event *B*.

2.2.7.2 Bayesian Inference for Uncertainty Quantification in Oil Recovery Estimation

Bayes' theorem can be used to quantify the uncertainty of simulation models in predicting recovery from a reservoir. Considering equation 3 and given the historical data (*B*), we can incorporate different parameterizations of the reservoir (*A*) into Bayes' formula, and calculate (update) the posterior probability of the model parameters. Since, oil recovery prediction is a continuous problem, the Bayesian framework can be written as:

$$p(m \mid O) = \frac{p(O \mid m)\, p(m)}{p(O)} \quad (4)$$

where $p(O)$ is given by equation 5:

$$p(O) = \int_M p(O \mid m)\, p(m)\ dm \quad (5)$$

In equations 4 and 5, *M* is the model, *m* is a vector of model parameters, *O* is a vector of the observed data and $p(m)$ is the prior probability distribution. $p(O|m)$ is called the likelihood of the data and can be defined as the expression of the probability of the

observation *O*, given the parameters *m*. The likelihood should assign a weight to a model based on the quality of the fit of model predictions to observed data. The model likelihood is evaluated by comparing the simulated model solution (reservoir simulation) with available observations (i.e. history matching). Minimizing the objective function (maximizing the likelihood) by sampling a number of possible reservoir descriptions from the prior, we update our beliefs about a given set of models. The likelihood *p(m|O)* describes the probability of the model given the data, i.e. the measure of to what degree the observed and modeled data differ. Hence, it is directly related to the minimized objective function via the likelihood model. For instance, it is common to assume the log of likelihood is equal to the negative objective function. The later characterizes how well the simulations fit the observed data. The goodness of fit is, often, evaluated by the square difference between observations and simulations normalized by double squared errors (inverse covariance matrix). This definition of the objective functions together with it's relation to the negative log of the likelihood assumes Gaussian statistics of errors. However, the choice of the objective function may vary depending on the optimization study task; some examples will be shown in the case studies section.

The form of likelihood depends on how we model uncertainty in the measurement. A Gaussian distribution is generally a good generic choice. Assuming the data measurement errors are normally distributed (Gaussian) around zero with a variance σ^2 at any given time and there is no simulation error, the probability that the true value of observed data is equal to simulated value is:

$$P(O_t \mid m) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\frac{(Obs - Sim)_t^2}{\sigma^2}\right\} \tag{6}$$

where t is the time step and σ is the standard deviation of errors. Assuming measurement errors are independent at each time step, the likelihood of the model is obtained from the product of the probabilities of individual measurements

$$P(O \mid m) = \left(\frac{1}{\sigma\sqrt{2\pi}}\right)^N \prod_{t=1}^{N} \exp\left\{-\frac{1}{2}\frac{(Obs - Sim)_t^2}{\sigma^2}\right\} \tag{7}$$

where N is the number of available data points. Equation 7 can be re-written as the following, considering $(\frac{1}{\sigma\sqrt{2\pi}})^N$ to be constant:

$$P(O \mid m) \propto \exp\left\{-\sum_{t=1}^{N} \frac{(Obs - Sim)_t^2}{2\sigma^2}\right\} \tag{8}$$

If we assume a least square misfit definition as:

$$M = \sum_{t=1}^{N} \frac{(Obs - Sim)_t^2}{2\sigma^2} \tag{9}$$

Then the likelihood function is:

$$P(O \mid m) \propto e^{-M} \tag{10}$$

Bayes' theorem relates posterior probability *p(m|O)*) with prior probability, *p(m)* with the likelihood *p(O|m)*. Bayesian *inference* provides a way of evaluating the posterior probability *p(m|O)* of multiple models generated using evolutionary optimization. Multiple good-fitting models generated by evolutionary algorithms models are highly likely (large *p(O|m)*) but are not equally probable. Their posterior probability *(p(m|O))* is computed numerically by Markov chain Monte Carlo (McMC) integration. An ensemble of models can be used to quantify the uncertainty of predictions

The end result of modeling in a Bayesian formalism is a posterior distribution. The probabilities obtained from Bayesian inference are referred to as posterior probability and are conditional on the observed data (for example pressure or production rate measurements). Many works have used the Bayesian framework, for example for estimating original gas in place [Aprilia et al. 2006], predicting of geological parameters [Glimm et al. 2001] and choosing between different exploitation scenarios for gas fields [Galli et al. 2004].

2.3 The Complete Framework

Now that we have reviewed a number of different algorithms for history matching and the approaches used for uncertainty quantification, we can propose a complete framework and finalize the workflow already describe in this chapter. Figure 10 presents the workflow that will be used throughout this thesis for history matching and uncertainty quantification.

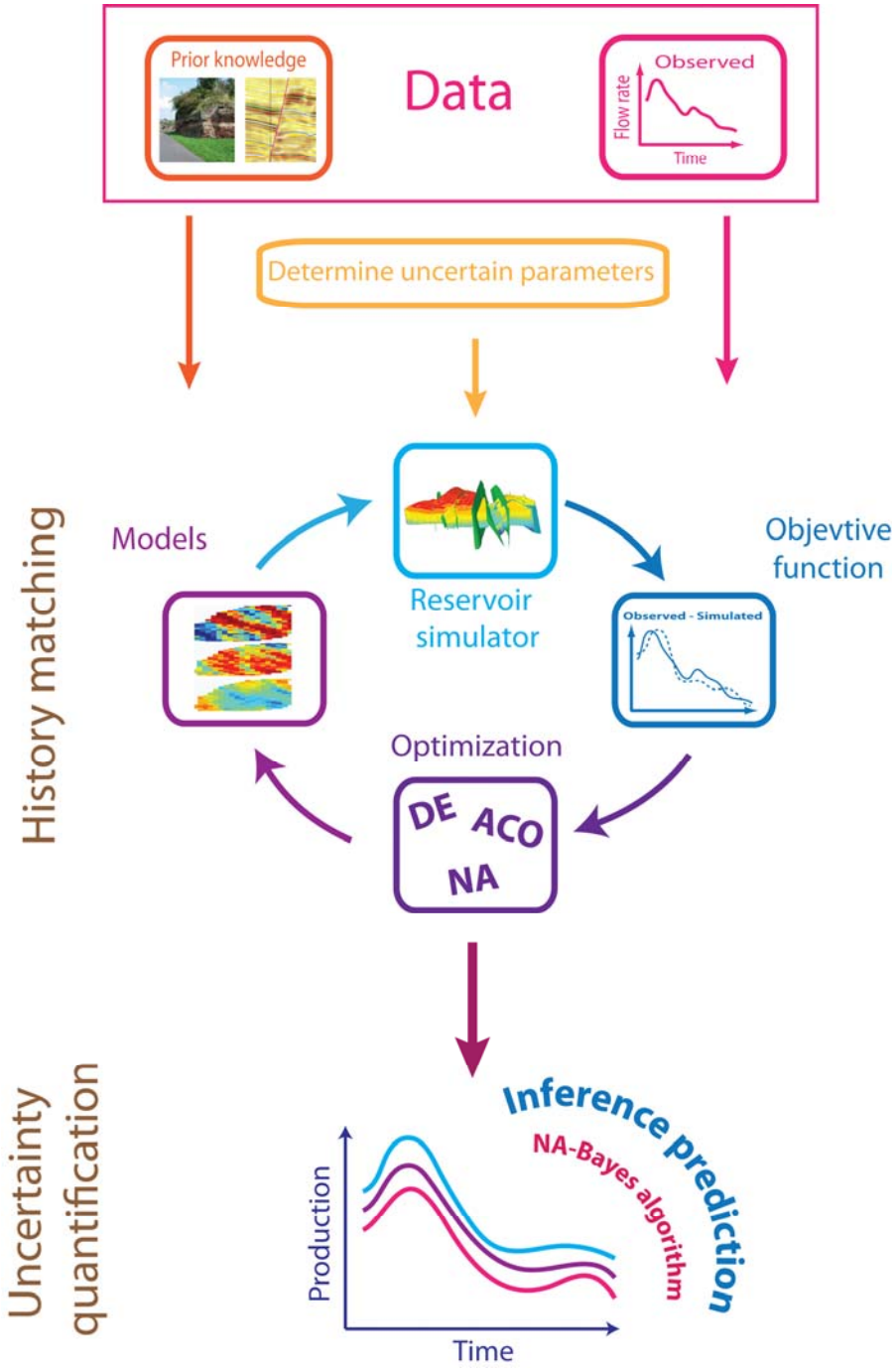

Figure 10: The complete history matching and uncertainty quantification framework used in this thesis

Our understanding of the modeled reservoir is based on data. By data we mean prior knowledge, which is used to build mathematical model relationships, and observations, which reflect the true but actually unknown behavior of the system subject to measurements uncertainty.

Our prior beliefs set a range of model definitions described by parameters or scenarios. From these beliefs we parameterize the reservoir description and set prior probabilities for these parameters. Thus, a mathematical/statistical model of a petroleum reservoir describes the distribution of the porous medium properties, which can be defined by geological body types, spatial correlation range, etc. Multiple models obtained using agent-based algorithms are actually sampled from our prior beliefs described by the probability distribution.

Next, the uncertain parameters and their prior ranges are determined. With this, we enter the history matching loop, where multiple reservoir models are generated using agent-based optimization algorithms. The mismatch between observed and simulated data is measured via a standard misfit definition. After generating an ensemble of models, these models are submitted to the inference step where the uncertainty of predictions in reservoir performance will be quantified.

In the next chapter, various algorithms used in this thesis within the above framework will be discussed. Ant colony optimization, differential evolution and Neighbourhood Algorithms as the engine for generation of multiple reservoir models will be described. The Neighbourhood-Bayes (NAB) routine as the uncertainty quantification tool will be also covered in next chapter.

Chapter 3

> "Attempt the end, and never stand to doubt. Nothing's so hard but search will find it out."
>
> Robert Herrick, English Poet (1591-1674)

Ant Colony Optimization, Differential Evolution, Neighbourhood and NA-Bayes Algorithms

This chapter is introduces of the algorithms used in this thesis. We review Ant Colony Optimization (ACO), Differential Evolution (DE), Neighbourhood Algorithm (NA) and NA-Bayes in the following sections and describe their background, working mechanisms and brief review of the applications.

Evolutionary and Agent-Based Stochastic Computation

Any method can solve a search problem given infinite time. This does not demonstrate intelligence; intelligent search is resource limited. In artificial intelligence, the Evolutionary Algorithm (EA) is the umbrella term for all computational models that are inspired by evolutionary mechanisms: reproduction, mutation, recombination, and selection. Although the particular representations can differ significantly from each other they all share basic principles. Every algorithm organizes a population of individuals.

3.1 Ant Colony Optimization

Nature has generously gifted many ideas to us for solve our problems. Social insects like ants which live in colonies are an example where we have learned how to solve difficult optimization problems based on their behavior when they are searching for their food. While the visual capabilities of ants are only developed rudimentarily, they

can find short paths from their nest to the food source. They deposit pheromone, an odorous chemical substance on their paths and this is used as a communication tool with other ants. These can smell the pheromone and will probabilistically choose the shorter paths which have been marked by stronger pheromone density. This complex behavioral pattern has encouraged scientists to investigate how ants can accomplish their tasks which can far exceed their individual capabilities.

3.1.1 Early Experiments

Goss et al. [1989] performed a simple experiment to study how ants can find the shortest path from their nest to the food source. They used a double bridge connecting a nest of Argentine ants to a food source. The two branches of this bridge had different lengths with the long one being twice as long as the short branch. Figure 1 shows a schematic representation of this experiment.

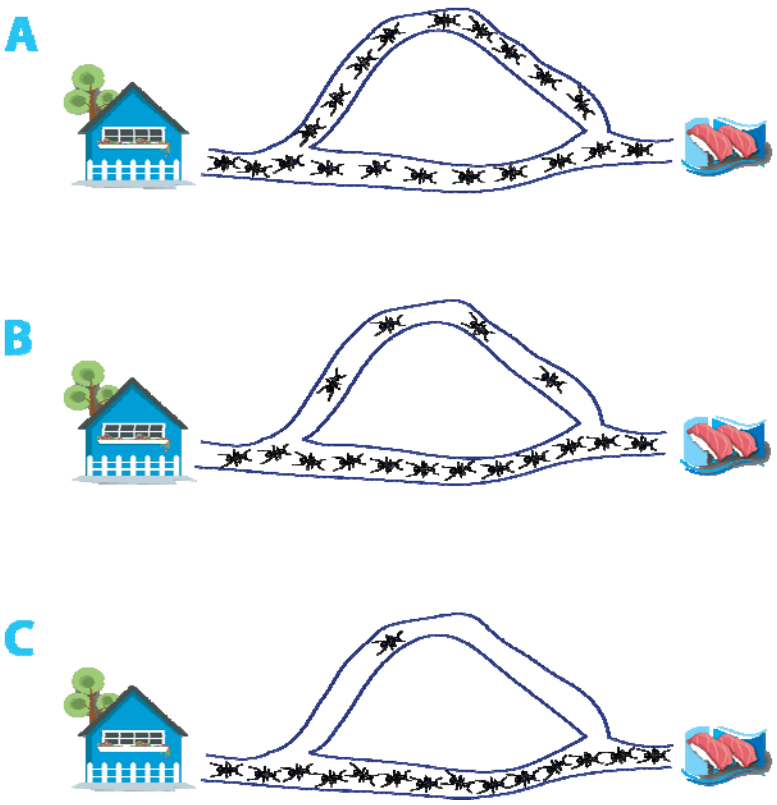

Figure 1: Schematic diagram of double bridge experiment

In the beginning, ants were allowed to freely move between the nest and the food source. They observed that initially two branches appear to have the same length and

ants randomly choose one of them (section A in figure 1). After some time the percentage of ants using the shorter path increased (section B in figure 1) and finally almost all the ants used the shorter path to get to the food source (section C in the figure 1). It is interesting to note that even at the end, a very small number of ants may still choose the longer route which was interpreted as path exploration.

The first returning ants were the ones who chose the shortest path to the food source. This shortest path was visited twice in the tour going from nest to the food source and return from source to the nest. Individual ants deposited pheromone both in their forward and backward movements and other ants can smell this pheromone. When the next ants wanted to choose their path, they make a probabilistic decision based on the pheromone density on that particular path. Because more ants are coming from the shorter path, the density of pheromone on this path is larger than the longer route. This pheromone density difference increases the chance of selecting the shorter path for the next ants. Continuing this pattern, most of the ants were stimulated to follow the shorter path at the end.

The above mentioned experiment proved that ants have their own built-in optimization capability. These experiments showed that the key to the success of the ants in doing complex tasks was indirect communication or *stigmergy* between members of the colony. This has been an inspiration for researchers to develop ant colony optimization algorithms which is described in next section.

3.1.2 From Real Ants to Ant Colony Optimization

Ant colony optimization was first introduced by Dorigo [1992] to solve the travelling salesman problem. In the travelling salesman problem the goal is to find a closed tour between N cities with the minimal length. Dorigo used the ant system for solving this problem and later proposed another algorithm called the ant colony system [Dorigo and Gambardella, 1997] to improve the performance of the original ant system. The basic idea behind all ant-based algorithms is using a positive feedback mechanism to reinforce good solutions based on the analogy with pheromone trail laying behavior of some species of ants [Dorigo et al. 1991]. In order to illustrate the basic mechanism

underlying ACO, simple ant colony optimization (S-ACO) [Dorigo and Stutzle, 2004] is described in the next section.

3.1.3 Simple Ant Colony Optimization: S-ACO

The simple ant colony optimization (S-ACO) for finding the shortest path on a graph $G = (N, A)$ works in the following way. Each artificial ant (ants throughout the rest of this thesis) starts from a source node in the graph which is illustrated in figure 2. While ants move forward towards the destination, they build a solution by choosing the next node to continue their tour from their current location on the graph. This selection of node is done probabilistically from the available neighbor nodes. One should note that in the graph $G = (N, A)$, two nodes are considered to be neighbors if there exists an arc $(i, j) \in A$. Each arc of the graph has been assigned a variable τ_{ij} called the artificial pheromone trail (pheromone through the rest of this thesis). Initially the amount of pheromone trail is the same for all of the arcs. The probability of selection is controlled by artificial pheromone density, with the choice being biased toward the paths marked by stronger pheromone deposited in previous tours of the ants in graph.

Considering the node i and the ant k, the pheromone trail is used in the following way to compute the probability of selecting the node j, for the next step of tour:

$$p_{ij}^{k} = \begin{cases} \dfrac{\tau_{ij}^{\alpha}}{\sum_{l \in N_i^k} \tau_{ij}^{\alpha}} & \text{if } j \in N_i^k \\ 0 & \text{if } j \notin N_i^k \end{cases} \qquad (1)$$

N_i^k is the neighborhood of ant k when in node i. N_i^k contains all the nodes connected by an arc to the current location of the ant in the graph except the previous node which that ant has visited. In this way, the ant will not return to its previous location. The only exception to this rule is when an ant gets to a dead end, where there is no path to move forward, so she can return to the previously visited node. The neighbor nodes for decision point i are marked with pink color in the figure 2.

In the S-ACO algorithm ants do not deposit pheromone while they are in the forward mode (walking toward the destination). This is required to avoid forming loops,

considering the deterministic backward movements of ants while they are heading back from the food source to the nest. Ants memorize the path they followed in order to reach the destination and are able to retrace it using an explicit memory. While moving backward, ants deposit pheromone on the arcs of the graph which they have already visited.

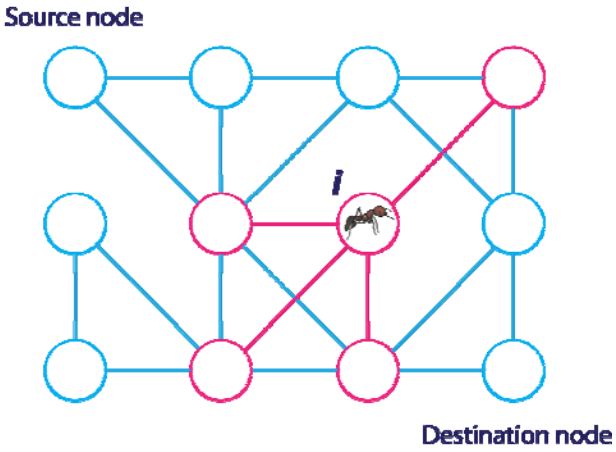

Figure 2: Simple ant colony optimization, S-ACO

3.1.3.1 Pheromone Update

After reaching their destination, ants start to return to the source node and trace the memorized path which they have followed to get to the destination. Due to different path length, the time frame for the ants returning to the source node will be different and the ants who have chosen the shorter routes will return faster. During their return trip, ants deposit an amount of pheromone on the arcs they followed. The pheromone value on the arc *(i, j)* changes when ant *k* deposits the pheromone according to the following rule.

$$\tau_{ij} = \tau_{ij} + \Delta\tau^{k} \qquad (2)$$

This pheromone update on the route increases the chance of choosing the same arc when the next ants visit that path. Choosing the value for the amount of pheromone deposited by ants $\Delta\tau^{k}$ is an important step. This value can be constant in that the ants visiting shorter paths can deposit their pheromone faster than colleagues who have visited longer routes, or its value can be a function of path length in which the shorter path will receive more pheromone.

3.1.3.2 Pheromone Evaporation

Pheromone evaporation helps the ants to forget poor quality results. A pheromone evaporation rule is defined in S-ACO in order to simulate this action in real ants. Pheromone is evaporated according to the following equation:

$$\tau_{ij} = (1-\rho)\tau_{ij} \qquad (3)$$

where $\rho \in (0,1]$ is the pheromone evaporation rate. Pheromone evaporation completes one iteration of S-ACO. It is interesting to note that in real ants, pheromone evaporation is too slow and has no effect on the search process, but in the case of artificial ants, especially considering large optimization problems, pheromone evaporation plays a very important role [Dorigo and Stutzle, 2001].

Based on the above discussion, any ACO approach must define the following parts for a new problem:

1. Transition rule: A heuristic function which guides the search with problem specific information
2. Pheromone deposition rule: A trail definition which states what information is to be stored and allows ants to share information about good solutions
3. Pheromone trail update rule: An update rule that defines the way in which good solutions are reinforced.

There are many other ant-based methods which use more advanced strategies for pheromone deposition, evaporation and the way ants complete their search tours. For a good review on these methods interested readers may refer to [Dorigo and Stutzle, 2004] and [Blum, 2005].

3.1.4 Applications of Ant Colony Optimization

There are numerous examples where ACO has proven to be a very efficient method for solving optimization problems. In computer science, Al-Ani [2005] used ACO for pattern classification problems. Rajpoot [2004] applied ACO for image coding. Wang

[2009] used ACO for optimization of mobile networks. Medical sciences have also benefited from ant colony optimization. Oliver et al. [2006] used ACO to optimize structure based drug design. Ant colony optimization has also been coupled with other intelligent optimization methods to solve challenging problems. Examples include using a hybrid of ant colony optimization and particle swarm optimization for function optimization [Shelokar et al. 2007], a hybrid of ACO and genetic algorithm for optimization of multiple sequence alignment in computational biology [Lee et al. 2008] and a hybrid system with fuzzy logic to form a case-based reasoning system [Kou and Chen, 2005].

In engineering problems, as an example, ACO has been successfully applied for designing electrical power systems [Ouiddir et al. 2004] and GPS networks [Saleh, 2002]. Murugappan [2005] formulated the drilling sequence problem in automatic drilling machines in terms of the travel salesman problem and solved it using an ant-based optimization method. Abbaspour [2001] used ACO for estimating unsaturated soil hydraulic parameters. Li and Hilton [2005] applied ACO for minimizing the number of locations required for spatial sampling in long term water monitoring networks. Leijen and Hermand [2006] used an ant system for inversion of geoacoustic properties of a shallow water environment. Zhang [2005] used ACO for dynamic optimization of chemical processes. ACO has also been applied to optimization of natural gas pipeline [Chebouba et al. 2009].

In petroleum engineering, there is limited work on the application of ant colony optimization. Pedersen et al. [2002] proposed an approach for extracting and interpretating faults from 3-D seismic data, called ant tracking. This algorithm works with a discretized volume of space (voxels). Ant tracking has been implemented in the Petrel commercial software [2009]. In their approach, ants are put as seeds in the seismic discontinuity volume. Each ant has a territorial radius: no other ant is placed within this radius. Ants start looking for faults by deploying pheromone in the paths that look more similar to the real fault. In this way true fault information attracts many ants and data from reflections and other noises do not get many ants. This gradually leads to an increased level of detail, faster interpretation of fault surfaces from fault attributes and an earlier understating of fault systems.

Recently Razavi and Jalali-Farahani [2008] highlighted the opportunity from using ant colony optimization algorithms in petroleum engineering problems and applied the CACO algorithm [Jayaraman et al. 2000] to the estimation of porosity and permeability in the well flow pressure equation:

$$P_{wf} = P_i - \frac{q\mu}{4\pi kh}[\ln\frac{kt}{\phi\mu c r_w^2} + 0.809] \quad (4)$$

They first obtained the well pressures using known values for both porosity and permeability and then estimated these values (2 unknown variables) using the CACO algorithm in order to obtain the pressure profile.

Other examples of the application of the ACO algorithm in various areas are explained in [Dorigo, 2004, 2006, 2008].

3.1.5 Ant Colony Optimization for Continuous Problems

Originally ACO was designed for discrete optimization problems. However, many researchers have attempted to extend this approach in order to handle continuous-variable optimization problems. Continuous problems cannot be represented in a graphical form where nodes are connected by edges which can be marked by artificial pheromone.

Early ideas were to use some sort of search space discretization. Bilchev and Parmee [1995] introduced one of the first extensions of ACO to solve continuous problems and called their method continuous ACO (CACO). They used a finite set of regions in each iteration. These regions were randomly explored by ants within a radius of exploration. These ants start their tour from the nest which was randomly sited in the search space. Eventually better paths were selected by pheromone trail diffusion, evaporation and recombination.

Wodrich [1996] proposed another algorithm for handling continuous optimization problems. In his proposed approach, two different types of agents take the responsibility

for searching the parameter space. Local search agents which make about 20% of the population do a similar task to the method introduced by Bilchev and Parmee [1995]. The remaining 80% of the population are global agents which do random walks to find new regions that contain good solutions.

Another ant-related approach which was proposed for tackling continuous optimization problems is the API algorithm [Monmarche et al. 2000]. API can handle both discrete and continuous variables in optimization. Other ant-inspired algorithms which have been proposed for continuous optimization are: Continuous Interacting Ant Colony (CIAC) [Dreo and Siarry, 2004], continuous orthogonal ant colony (COAC) [Hu et al. 2008] and Continuous Ant Colony System (CACS) [Pourtakdoust and Nobahari, 2004], Direct Ant Colony Optimization (DACO) [Kong and Tian, 2006] and Charged Ants for Continuous Dynamic Optimization (CANDO) [Tfaili and Siarry, 2008].

Recently Socha and Dorigo [2008] proposed a new continuous ant colony optimization algorithm called ACO_R which has been used for history matching in this thesis. In the next section this algorithm is described in more detail. As discussed in [Socha 2008], although CACO, API and CIAC algorithms are inspired by ants behavior but they have conceptual differences with the original ACO framework and thus they do not qualify as extensions of ACO. In this sense ACO_R is the closest method to the spirit of ACO for discrete problems.

3.1.6 Ant Colony Optimization for Continuous Domains - ACO_R

The heart of ACO_R is a solution archive with k models which keeps track of the solutions (figure 3, section A). In this archive, solutions are ranked based on their quality, which means models with lower misfit values get a higher rank and will be at the top of the table.

There are k rows and n columns in the archive, where k is the number of models that are kept in the archive and n is the number of dimensions of the problem. Each row of this archive (for example s_l) stores the parameter values, the computed misfit, and weights for a single model. The i^{th} unknown parameter value of the l^{th} model is denoted by s_l^i. Each column in this table represents one of the parameters or dimensions of the

problem. The objective function value for each model is denoted by $f(s)$. The next item in this archive is the weights of the solutions which we denote by w. These weights are a function of solution quality (Eq. 5) and will be used to probabilistically build new solutions. The models which result in a better solution (lower misfit) get higher weights and will be placed on the top of the archive. A single row of this solution archive is illustrated in figure 3, section B. Model l is highlighted by the dashed pink oval in section (A) of the figure 3. It is one of the solutions for a history matching problem, with 5 unknowns, including porosity, horizontal and vertical permeabilities, rock compressibility and fault throw.

Like almost every evolutionary algorithm, initially the archive is filled with random solutions and the fitness of each model is evaluated. If the number of ants evaluated at each iteration of the ACO_R algorithm is m, then at each iteration of the optimization, m generated solutions are added to the population and from the solution archive which now contains $k+m$ models, the m worst solutions are removed to keep the archive size fixed. This action simulates the pheromone update part in the discrete ACO. The idea of keeping the archive size fixed is similar to the steady state genetic algorithm in which also the population size remains the same during the optimization run. The remaining models in the archive are sorted according to their misfit score. The aim is to bias the search process towards the good regions of the search space which contain low misfit models by probabilistically constructing new solutions.

To construct a single model, at each step ($i=1, 2 \ldots n$) an ant chooses a value for each unknown parameter in the problem. This is done by random sampling of the probability density function (PDF). We use a mixture of Gaussian kernels to represent the PDF (figure 3, section C). The components of the solutions are used to dynamically generate probability density functions and modify their shape. The weight of each member in the archive is used to probabilistically select the members of archive. It is computed based on the following equation:

$$\omega_l = \frac{1}{qk\sqrt{2\pi}} e^{-\frac{(l-1)^2}{2q^2k^2}} \qquad (5)$$

where k is the size of the archive, l is the rank of the solution and q is a parameter of the algorithm. In case the value of q is small, the best-ranked solutions are strongly preferred, and in case it is larger, the probability of selecting models in the archive becomes more uniform. The parameter q controls the balance between exploration and exploitation in ACO_R.

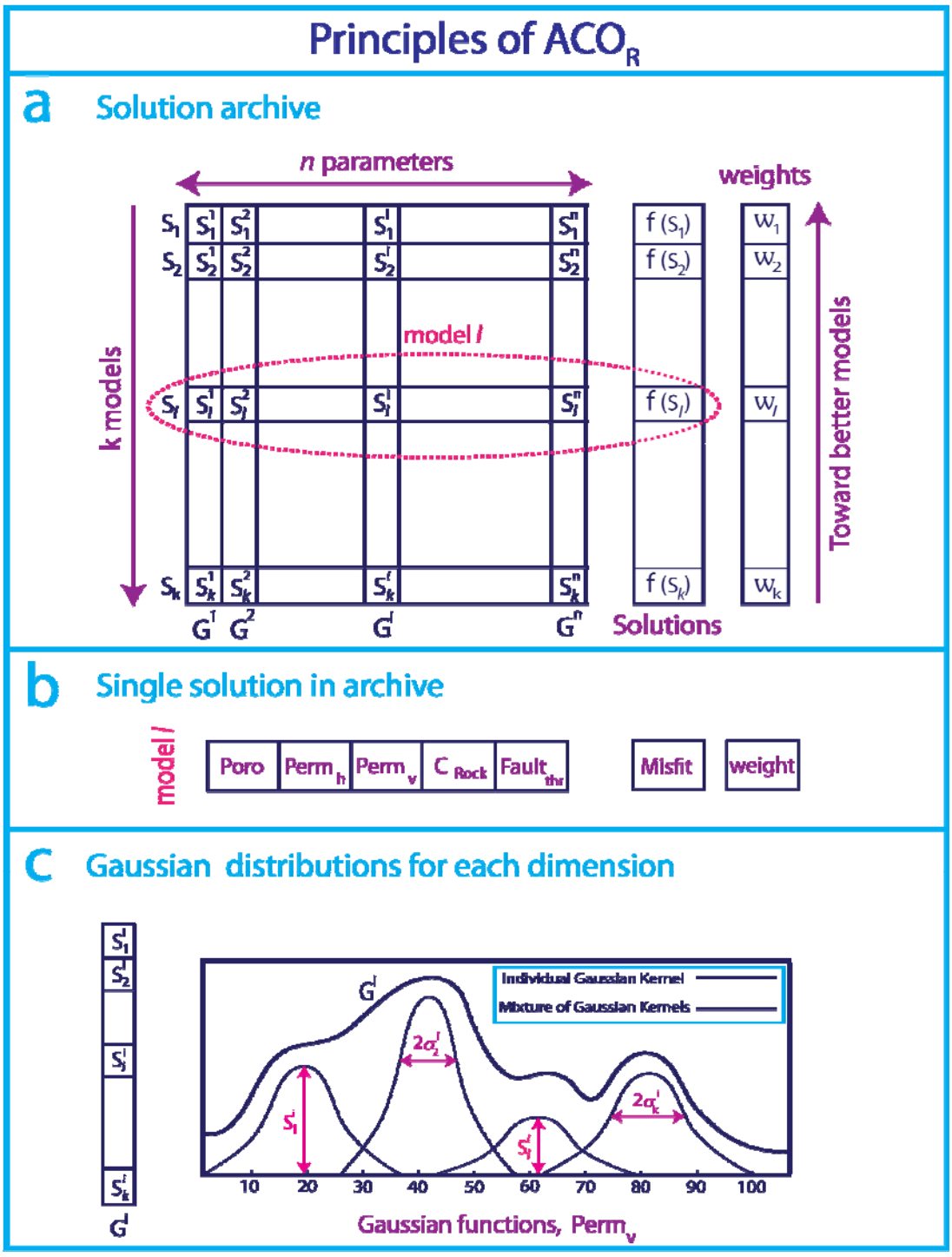

Figure 3: Principles of ant colony optimization for continuous domains (ACO_R)

In order to generate new solutions, first the ants choose one of the solutions in the archive according to the following probability:

$$p_l = \frac{\omega_l}{\sum_{r=1}^{k} \omega_r} \tag{6}$$

The ACO_R selection strategy differs from the simple GA selection methods like roulette wheel because the probability of selection is not a direct function of the fitness, but it is computed indirectly from the fitness considering the effect of parameter *q*. Individual Gaussian kernels (figure 3, section C) are characterized by the vector of means ($\mu^i = \{s_1^i, ..., s_k^i\}$) and standard deviations given by the following equation:

$$\sigma_l^i = \xi \sum_{e=1}^{k} \frac{\left| s_e^i - s_l^i \right|}{k-1} \tag{7}$$

The standard deviation value represents the average distance between the i^{th} variable of the selected solution and the i^{th} variable of the other solutions in the archive. It is multiplied by a parameter $\xi > 0$ which plays the role of pheromone evaporation rate here. With lower ξ values, the search is less concentrated on the previously visited areas of the search space; hence the convergence speed of the algorithm will be higher. The mixture of Gaussian kernels is the weighted sum of the individual Gaussian kernels and is computed for each dimension of the problem with the following equation:

$$G^i(s) = \sum_{l=1}^{k} \omega_l \frac{1}{\sigma_l^i \sqrt{2\pi}} e^{-\frac{(s-\mu_l^i)^2}{2\sigma_l^{i2}}} \tag{8}$$

The mixture of Gaussian kernels will guide the ants through their search process. In the next step we sample the chosen Gaussian kernel to obtain the new models for the next iteration of the algorithm. This process is continued for all dimensions of the problem by each of the *m* ants, and at each iteration of the algorithm, until the search process converges to the optimal value.

3.1.6.2 Handling Constraints in ACO_R

ACO_R assumes that the decision variables are unconstrained. This assumption and using normal Gaussian functions to sample the parameter values sometimes leads to unacceptable values for parameters when ACO_R is used in history matching. In order to handle constraint optimization problems, Socha [2008] proposed to return infinite values for the objective function when the parameter values violate the constraints. Since the solutions are ranked based on their objective function value, in this way feasible solutions will always be better than the solutions giving values outside the predetermined constraints. This approach together with other penalty methods including returning the boundary values of the parameters, did not give satisfactory results when applied to the history matching problems. We replaced the original normal distribution in the ACO_R with a two-sided truncated Gaussian distribution [Robert, 1995] in order to control the constraints for variable ranges.

3.1.6.3 Applications of ACO_R

Socha [2008] successfully applied ACO_R to the optimization of several mathematical functions; but the success for application of ACO_R are not only limited to this area. It has also achieved good results in optimization problems in other fields. Examples in computer science problems include its application to the training neural networks [Blum and Socha, 2005] and determining fuzzy logic membership functions [Jiang et al. 2008].

ACO_R has also helped weather forecasting. Zhang et al. [2007] proposed a method for fast typhoon eye tracking based on a continuous ant colony optimization algorithm.

Socha [2008] used ACO_R for mixed variable optimization problems in mechanical engineering. He demonstrated the application of his algorithm with three examples, pressure vessel design, coil spring design and thermal insulation system design. All of the three examples involved both discrete and continuous variables. Madadgar and Afshar [2009] applied ACO_R to the capacity optimization of a hydropower plant. They also proposed some modifications to the original ACO_R algorithm to improve its performance. Schluter et al. [2009] used a modified version of ACO_R for integrated process and control systems in wastewater treatment plants.

3.1.6.4 Comparison of ACO_R with other Population-Based Methods

Socha and Dorigo [2008] made a comparison between the performance of ACO_R and two other groups of optimization algorithms. The first group is classified as continuous ant colony optimization algorithms and include CACO, API and CIAC algorithms. The comparison for this group was done using a test bed of 10 functions. The second group is known as evolutionary algorithms and other metaheuristics adopted for continuous optimization problems. This group consisted of the following algorithms: (1+1) ES as the simplest form of evolutionary strategies where one parent generating one offspring at each iteration [Kern et al. 2004], Evolutionary Strategy with Cumulative Step Size Adaption (CSA-ES) [Ostermeier et al. 1994], Evolutionary Strategy with Covariance Matrix Adoption (CMA-ES) [Hansen and Ostermeier, 2001], Mixed Bayesian Optimization Algorithm (MBOA) [Ocenasek and Schwarz, 2002], Iterated Density Estimation Algorithm (IDEA) [Bosman and Thierens, 2002], Continuous Genetic algorithm (CGA) [Chelouah and Siarry, 2000], Enhanced Continuous Tabu Search (ECTS) [Chelouah and Siarry, 1999] and Enhanced Simulated Annealing (ESA) [Siarry et al. 1997]. Socha used 10 functions to compare the performance of ACO_R with (1+1) ES, CSA-ES, CMA-ES, IDEA and MBOA algorithms. He also used 15 test functions for comparison between ACO_R algorithm and CGA, ECTS, ESA methods.

This study revealed the following conclusions. ACO_R is the clear winner when it is compared to other ant colony optimization algorithms. ACO_R outperforms CACO, API and CIAC algorithms in all of the 8 functions used as the benchmark. For (1+1) ES, CSA-ES, CMA-ES, IDEA and MBOA algorithms, ant colony optimization for continuous domains (ACO_R) found the best value in 4 out of 10 cases. When it was defeated by CMA-ES in 5 of the cases, the result of the ACO_R algorithm was slightly worse than CMA-ES. It is interesting to note that CSA-ES, IDEA and MBOA did not obtain the best results for any of the 10 functions used for testing and (1+1) ES was only successful for one of the functions. When ACO_R was compared with CGA, ECTS, ESA, it could find the best solution in 5 functions out of the 15 tests. CGA was the best performing in 4 functions, while the difference between the results obtained by CGA and ACO_R was small. ECTS was good at finding the minimum function value for 4 of the test functions. Again the results of ACO_R were not much worse than the solutions

obtained by ECTS. ESA was the only algorithm which was unsuccessful in all the test problems used in this comparison. Differential evolution approach was tried for only 3 functions and managed to locate the best value for one of them.

Socha [2008] also compared the results of ACO_R for solving three different mixed variable problems in mechanical design with 9 different optimization methods used for these problems. These approaches include nonlinear integer and discrete programming (NLIDP) [Sandgren, 1990], mixed integer-discrete-continuous programming (MIDCP) [Fu et al. 1991], sequential linearization approach (SLA) [Loh and Papalambros, 1991], nonlinear mixed discrete programming (NLMDP) [Li and Chou, 1994], genetic algorithm (GA) [Wu and Chow, 1995], evolutionary programming (EP) [Cao and Wu, 1997], evolutionary strategy (ES) [Thierauf and Cai, 2000] and combined heuristic optimization approach (CHOPA) [Schmidt and Thierauf, 2005]. In all of the above cases, ACO_R obtained better results for the design optimization problem and in some of the cases this was achieved from fewer function evaluations.

In a recent work Kovarik [2006] compared the performance of ACO_R with particle swarm optimization (PSO) and a hybrid of PSO with genetic algorithm (GAPSO) for training and optimizing the performance of a neural network system. The goal was to minimize the output error from neural networks by adjusting the parameters of the network using optimization methods. He used seven real world datasets in this research and concluded that ACO_R outperformed PSO and GAPSO methods in the problems tested in this study.

These evaluations have shown that ACO_R performs very well in both continuous and mixed-variable problems and outperforms many optimizations methods which are currently in use.

3.2 Differential Evolution

Differential Evolution (DE) is a powerful global optimization algorithm and was introduced by Storn and Price [1995]. DE grew out of Price's attempts to solve the Chebychev polynomial fitting problem that had been posed to him by Storn. DE is a parallel agent-based search algorithm which uses N_p D-dimensional parameter vectors

as the population in each generation. Then it tries to evolve this population by simple arithmetic operations on these vectors to form new solutions to the problem.

Figure 4 illustrates the concept of building difference vector in differential evolution. In this figure, DE is applied to find the minimum of a unimodal function.

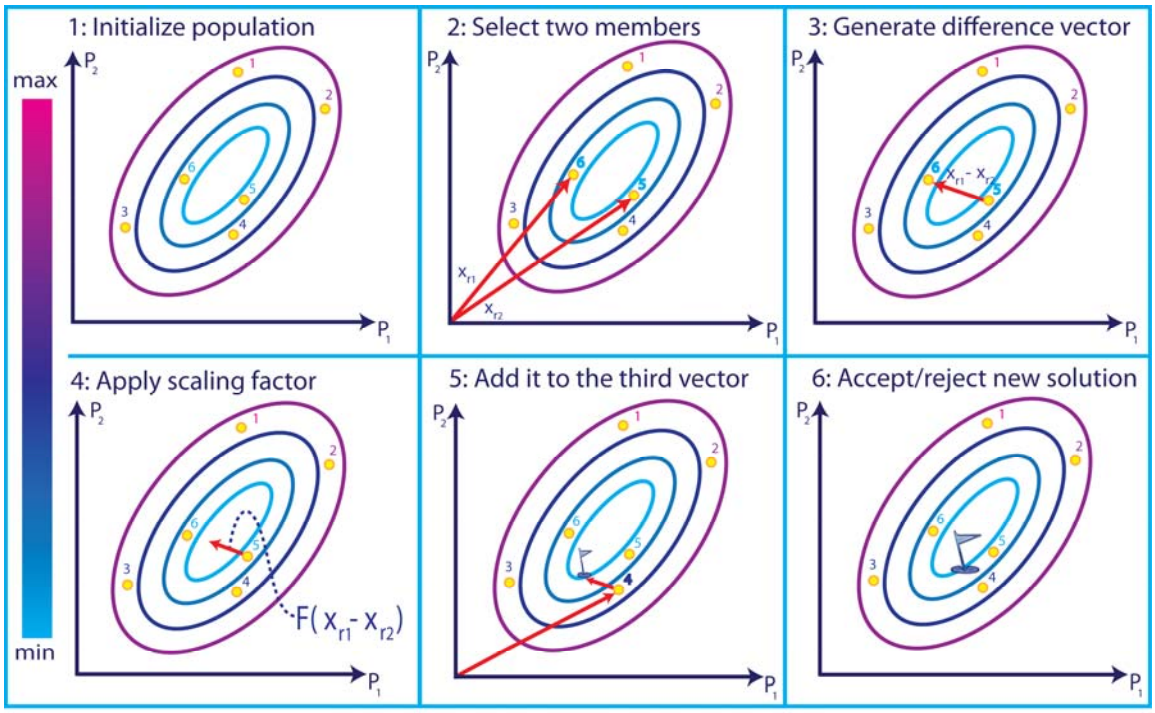

Figure 4: Illustration of differential evolution algorithm

Like any other evolutionary algorithm, DE starts with a population of N_p parameter vectors representing the candidate solutions. In the first step the population is initialized with 6 individuals randomly scattered in the search space. Each individual member in the population is a vector of real numbers. All vectors are uniquely indexed from 1 to N_p for bookkeeping. In the second step two vectors are randomly selected from the current population and in the next step (shown in part 3 of figure 4), the difference vector between two selected members is computed. In step 4, this vector is multiplied by a number called scaling factor. In step 5, we select another member (individual 4) in the population and we add the scaled difference vector which was obtained in step 4, to this new selected member. Finally we come to the selection step which is shown in part 6 of

figure 4. In differential evolution each trial vector competes against the population vector of the same index. Since this trial vector is the first new member generated, it is going to compete with the member indicated by index of 1. For the function shown in figure 4, because the trial vector has a lower objective function comparing to the individual number 1, it will be selected to proceed to the next generation. The above procedure is repeated for each individual in the N_p population to form the next generation of solutions.

In terms of an agent-based algorithm, DE has the following elements:

3.2.1 Differential Mutation

In DE, the perturbation to the models in each generation is made by adding a weighted difference between two (or more) other chosen models. This forms the differential mutation part in DE. To introduce a perturbation to each vector, in each generation *G*, a mutant vector is produced by:

$$v_{G+1} = r_{1,G} + F(r_{2,G} - r_{3,G}) \tag{9}$$

r_1 is the base vector and r_2 and r_3 are two other vectors chosen from the population from the range [1, N_p]. F is called scaling factor and is a real positive constant parameter that affects the differential variation between two vectors and controls the rate which population evolves. Usually in the literature this number is between 0 and 2. Depending on the value of the scaling factor value, the difference vector may become larger or smaller than its original size.

The operations of DE are different from the classical GA in at least two points. Traditional GAs work with logical operators on bit strings. The solutions in DE are vectors of real numbers and the algorithm works by arithmetic operators on these vectors. Also DE, in contrast with GA, selects at least two vectors at each step and the difference between these vectors is added to the third vector.

3.2.2 Crossover

To increase the population diversity, a crossover operation is performed after the mutation step in differential evolution. In this step the parent (donor) vector is mixed with the mutated vector to produce the trial vector. There are two types of crossover schemes for differential evolution – binomial and exponential.

In binomial crossover, a random number (rand (*j*)) between 0 and 1 is selected. Each element of the trial vector is replaced by its pair in the mutant vector if rand (*j*) is less than or equal to C_r. This process is shown by equation 10:

$$u_{J,G+1} = \begin{cases} v_{J,G+1} & if\,(rand(j) \le C_r) \\ x_{J,G} & if\,(rand(j) > C_r) \end{cases} \qquad (10)$$

where j = 1, 2, …, D and C_r is the crossover constant $\in$ [0, 1]. In binomial crossover, the number of components taken from the mutant vector has a binomial distribution [Price et al. 2005].

In exponential crossover, a random integer n is chosen from [1, D], with D being the number of dimensions (unknowns) in the problem. We start from this chosen point in the target vector to crossover (exchange) components with the donor vector.

$$u_{J,G+1} = \begin{cases} v_{J,G+1} & for\ \ j = \langle n \rangle_D, \langle n+1 \rangle_D, ..., \langle n+L-1 \rangle_D \\ x_{J,G} & for\ \ all\ \ other\ \ j \in [1, D] \end{cases} \qquad (11)$$

L is the number of components exchanged from the donor vector to the target vector and is selected from [1, D] according to the following pseudo code:

```
L=0;
WHILE (rand(j) < Cr AND (L< D))
DO
{
L = L + 1;
}
```

Crossover controls the number of components taken from the mutant vector. The probability that a component is mutated is determined by the mutation probability (p_m). The relationship between the crossover value and mutation probability is different for binomial and exponential crossover schemes and has been extensively studied by Zaharie [2007, 2009].

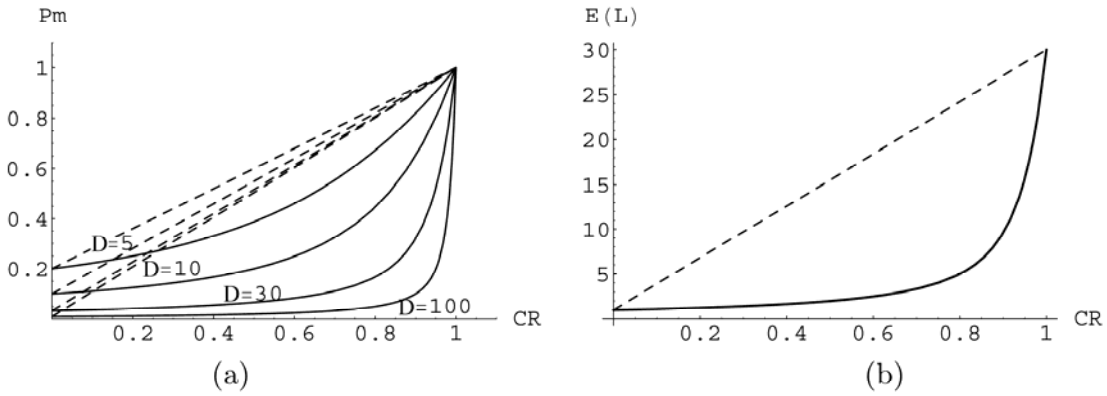

Figure 5: Effect of crossover rate (a) on the mutation probability and (b) on the average of the number of mutated components for D = 30 for binomial crossover (dashed line) and exponential crossover (continuous line) – taken from [Zaharie, 2007]

These studies showed that the relationship between mutation probability and crossover rate is linear for the binomial crossover scheme and nonlinear for the exponential crossover scheme. In this thesis the exponential crossover has been used for the differential evolution algorithm.

3.2.3 Selection

In DE, each trial vector competes with its parent and the fittest one survives to form the next generation.

To summarize the concept of differential evolution, we have the following pseudo code:

- Initialize population with N_p individuals
- Obtain objective function values
- Do until stopping criteria met:

- Select vectors
- Mutation
    - introduce perturbation
- Crossover
    - increase population diversity
- Selection
    - if new vector is better than parent

The main advantage of differential evolution is its simplicity. It has three control parameters, namely the population size, scaling factor and crossover rate. Recently some works have also proposed adaptive versions of differential evolution where tuning parameters are automatically and dynamically adjusted during the optimization [Liu and Lampinen, 2005], [Teo, 2005], [Brest et al. 2006].

3.2.4 Variants of Differential Evolution

Variants of DE are different in the way they perform mutation and crossover steps. In the literature usually different variants of DE are presented in the form of DE/*x*/*y*/*z*, where *x* is the vector that will be mutated, *y* specifies number of difference vectors used and *z* is crossover scheme. We have chosen to use four strategies of differential evolution in this work which are described in the next section.

3.2.4.1 DE/Rand/1

In this strategy, the base vector is chosen randomly and one weighted difference vector is added to it.

$$v_{i,G+1} = r_{1,G} + F(r_{2,G} - r_{3,G}) \tag{12}$$

In which r_1, r_2 and r_3 are all chosen randomly.

3.2.4.2 DE/Best/1

This works similar to the DE/Rand/1, but instead of selecting the base vector randomly, the algorithm selects the best vector with lowest cost function as the base and adds 1 vector difference to it.

$$v_{i,G+1} = r_{Best,G} + F(r_{1,G} - r_{2,G}) \quad (13)$$

Here $r_{Best,G}$ is the best individual in each generation and r_2 and r_3 vectors are selected randomly from current population.

3.2.4.3 DE/Rand to Best/1

This scheme's performance is between the DE/Rand and DE/Best since it uses a randomly selected vector and the best so far vector at the same time.

$$v_{i,G+1} = r_{1,G} + F(r_{Best,G} - r_{1,G}) + F(r_{2,G} - r_{3,G}) \quad (14)$$

3.2.4.4 DE/Best/2

This strategy works similar to DE/Best/1, but instead of one difference vector, two difference vectors are added to base vector. These two difference vectors are obtained from four randomly selected members in the population. The base vector is the best individual and two difference vectors are added to it. This can be represented by following equation:

$$v_i = r_{best,G} + F(r_{1,G} - r_{2,G} + r_{3,G} - r_{4,G}) \quad (15)$$

where r_1, r_2, r_3 and r_4 vectors are selected randomly from current population.

3.2.5 Comparison of Differential Evolution with other Optimization Methods

Storn and Price [1997] compared DE with two annealing methods: Annealed Nelder-Mead (ANM) and Adaptive Simulated Annealing (ASA). The comparison was done on a test bed of 10 functions and DE was the only search method that could find the global minimum of the all test functions. For the functions that other methods were also successful in finding the global minimum, DE had less function evaluations in comparison with ANM and ASA. They also compared DE-Rand strategy with two well-performing evolutionary algorithms. These were Breeder Genetic Algorithm and Evolutionary Algorithm with Soft Genetic Operators (EASY). DE performed very well

compared to these two methods and in 9 of 10 cases needed the least number of function evaluation.

Cruz et al. [2003] compared DE performance with two variants of the Breeder GA (simple and with sub-populations) for dynamic optimal control problem. The comparison was based on four criteria: 1) number of function evaluations 2) CPU time 3) performance index value 4) convergence efficiency (number of times that the algorithm found the global solution. The results indicate that although both DE and Breeder GA efficiency was 100%, DE was able to find a lower value for performance index value, with less function evaluations and with lower CPU time.

Joshi and Sanderson [1999] used DE for the minimal representation of multisensor fusion and compared the results of DE/Best/2 with GA (onepoint crossover and mutation operators for reproduction and stochastic universal sampling for natural selection). Results indicate that DE found smaller values of representation size from fewer number of problem evaluations than GA. In this study, DE solutions also had smaller interpretation errors.

Biesbroek [2006] used DE and GA for the optimization of three different interplanetary trajectory problems and showed the superiority of DE for this case. Chakraborti and Kumar [2003] compared simple genetic algorithm and island model of genetic algorithm for optimal scheduling of a strip mill. The island model genetic algorithm benefits from multiple populations by creating few islands each with few tribes. The results showed that among these three algorithms, the rate of convergence was fastest for DE and results were in general slightly better. Tvrdik [2006] also reported better performance of differential evolution in comparison with real-coded genetic algorithm.

Karaboga and Cetinkaya [2004] compared the performance of GA and DE for digital filter design and concluded that the performance of GA and DE were almost similar to each other in terms of least mean squared error, with DE obtaining slightly better value. From the point of view of convergence speed, DE converged significantly faster than GA to optimal region of search space. The performance of DE has also been compared with GA and simulated annealing (SA) for parameter estimation of diodes [Wang and

Ye, 2009]. Also for this problem, DE outperformed GA and SA both in obtaining lower objective function values and achieving a higher convergence speed.

Ursem and Vadstrup [2003] used DE for the parameter identification of induction motors. They compared the results of DE with results of their previous study where they used 8 different search algorithms for this problem. These algorithms include local search, simulated annealing, simple EA, diversity guided EA, evolution strategies with adoption of one σ, evolution strategy with adoption of both standard deviations and rotation angles, standard particle swarm optimization and diversity guided particle swarm optimization. The best performing algorithms among these 8 search methods were the two evolution strategy and diversity guided EA. They compared DE with Diversity Guided EA (DGEA), evolution strategies with adoption of one σ and an evolution strategy with adoption of both standard deviations and rotation angles. DE was able to locate the exact solution in all runs. Also the convergence speed of DE was significantly higher than all other search methods.

Vesterstrom and Thomson [2004] reported that differential evolution outperforms particle swarm optimization in most of the functions in a test bed of 34 functions, especially the functions where the problem is highly multimodal. Xu and Li [2007] also presented a comparison between PSO, DE and multi-parents crossover algorithm (MPC) and concluded that DE surpasses both PSO and MPC algorithms.

3.2.6 Differential Evolution for Engineering Problems

Differential evolution has been widely applied for solving engineering problems, for example electric and electrical engineering [Qing, 2009], aerospace engineering [Madavan, 2004], civil and urban engineering [Suribabu, 2010], electromagnetics [Qing and Lee, 2010], structural engineering [Savoia and Vincenzi, 2008], [Wu and Tseng, 2010], transport problems [Koh, 2007] and chemical engineering [Angira and Babu, 2006], [Babu and Munawar, 2007].

Despite the successful applications of differential evolution in many fields, there are a limited number of publications related to applications of this optimization algorithm for tackling petroleum engineering problems. Mekapati [2005] studied phase stability

problem in phase equilibrium using differential evolution. Henderson et al. [2010] applied differential evolution for calculation of critical points of thermodynamic mixtures.

Differential evolution has been used to history match coreflood data to estimate oil and water relative permeability and capillary pressure [Wang and Buckley, 2006]. In this example, differential evolution was coupled with a simple two-phase reservoir simulator to match oil recovery and pressure drop curves by tuning the unknown relative permeability and capillary pressure values.

Jahangiri [2007] applied differential evolution to optimize smart well operations to maximize oil production. Recently Zhang et al. [2009] used differential evolution for estimation of variogram properties in geostatistical frameworks

3.3 Neighbourhood Algorithm

The Neighbourhood Algorithm (NA) is a recent stochastic optimization method proposed by Malcolm Sambridge [1999 a]. It was originally developed for solving inverse problems in seismology.

The neighbourhood algorithm uses simple geometrical constructs called Voronoi cells (named after Georgy Voronoi 1868-1908) to find good regions of the search space. Voronoi cells divide the *n*-dimensional search space into separate regions. Each of these cells is the nearest neighbourhood region of the points, measured by a particular measure, usually the L_2 norm. Voronoi cell is a method to decompose the search space into *n* cells around *n* points by centering around the generated points.

3.3.1 Working Mechanism of Neighbourhood Algorithm

Figure 6 illustrates the working mechanism of the neighbourhood algorithm (NA). The NA starts the optimization by randomly generating n_{si} models in the search space and the objective function values are evaluated for these solutions. After this initialization step, the search space is portioned using Voronoi cells, each cell having a misfit value. These cells have unique shapes in each iteration which are determined by members of that generation. Then at each iteration, n_s / n_r new individuals will be generated in each

cell. Then the geometry of the old Voronoi cells will be modified to account for the newly generated models. For example, in figure 6, $n_s = 4$; so in each of the $n_r = 2$ cells, we generated 4/2 = 2 new models. This is followed by ranking all models according to their misfit score. From this ranking, the best n_r models with lowest misfit scores are chosen and then new n_s models are generated by a uniform random walk within Voronoi cells of these best n_r cells. It is important to note that the misfit values are assumed to be constant within each Voronoi cell. This procedure is repeated until a predetermined stopping criteria is met.

As the stopping criteria for NA, in this thesis the maximum number of function evolutions has been used to terminate the search. However it is possible to use other techniques to stop the algorithm or to re-start the search. These advanced methods can be useful in saving the computational resources when there is no further improvement in the performance of algorithm or when the algorithm is trapped in local minima. For example, Erbas [2006] introduced a monitoring procedure for NA to detect the iteration where NA is trapped in a local minimum and starts to over-refine that region. For this purpose, the average and minimum misfit obtained in each iteration are computed and if the improvement in these two criteria (Gen_{min} and Gen_{avg}) in two iterations are less than a threshold, that is considered as an indication that NA is trapped. Hadidi [2007] suggested using the following two criteria that both should be met to stop the search: 1) if for the past n_s models, there is no change in the Voronoi cell being sampled 2) The distance between the best-so-far obtained point and the new n_s models are less than a predefined number.

The geometry of Voronoi cells in NA allows the centre of sampling to change in different places, while simultaneously sampling from the best n_r regions. Also as stated by Sambridge [1999, a], we should note the best fit model in each iteration is not necessarily obtained by sampling the previous best fit Voronoi cell, but may have come from any other n_r cells. These two reasons help NA to simultaneously concentrate its sampling on different regions of the search space and produce a diverse set of models.

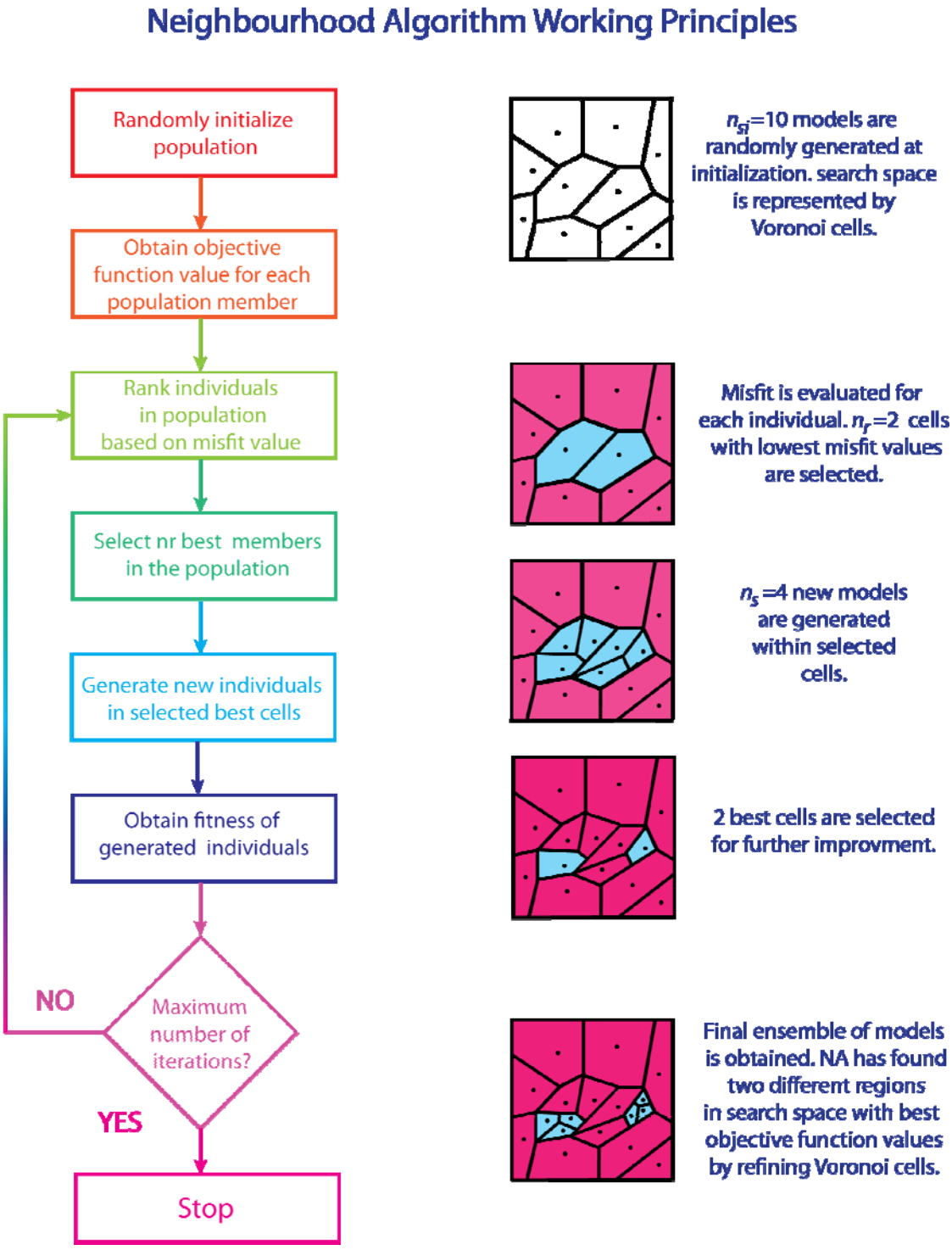

Figure 6: Working principles of Neighbourhood Algorithm (NA)

Sambridge [1999 a] states that NA search is only directed by the rank of the models according to their misfit values, not just misfit values themselves. This gives more

freedom to the user to define any misfit type [Sambridge 2003]. It should also be noted that although NA is simple in principle, it is not very easy to write the code of this algorithm. We use a parallel version of NA algorithm in this thesis.

3.3.2 Tuning Parameters of NA

In NA, only two parameters control the behavior of the algorithm: n_s and n_r. Sambridge states that the n_s/n_r ratio rather than the individual values of these two tuning parameters, control the amount of exploration and exploitation for NA [Sambridge, 1999 a]. There are several works to understand the behavior of the neighbourhood algorithm considering it tuning parameters. Elabed [2003] states that by keeping n_s/n_r ratio high one can achieve good exploitation of best fit regions and lower n_s/n_r ratios will result in good exploration of the parameter space. She reports that sampling is likely to get trapped in local minima if the first best models are all located in the same regions of the search space, forcing the algorithm to discard other regions for exploration. Sambridge suggests using $n_s/n_r = 2$ as a first trial. Despite many efforts to generalize suggestions for selecting tuning parameters of NA, it seems that this decision is problem-dependent. Reviewing different choices in a broad range of literature reveals that this choice is not simple at all. There have been reports using the maximum possible value ($n_s/n_r = 1$) in Beghein et al. [2002], Resovsky and Trampert [2002] and Beghein and Trampert [2004]. Many researchers followed the original suggestion to use $n_s/n_r = 2$ including Yoshizawa and Kennett [2002], Vallee and Bouchon [2004], [Agostinetti et al. 2004], Jansky et al. [2007], Yao et al. [2009] and [Edris, 2009]. Other works use higher values for n_s/n_r ratio like 4 in Litvak et al. [2005] and 5 in Oye and Roth [2003] and Kennett [2006]. Erbas [2006] used different values of n_s/n_r for each problem in her thesis including 1, 5 and 10.

Erbas [2006] also showed that selecting larger initial population and larger n_s values improves the exploration of the search space and this improves the performance of the neighbourhood algorithm; however it is obvious that the improvement comes at the cost of increasing the required misfit evaluation.

3.3.3 Improvements Introduced for NA

Wathelet [2008] points out that some inverse problems may require a search space with irregular search boundary conditions, whereas the original NA proposed by Sambridge is limited to a hyper-box [Alpern and Carter, 1991]. The irregular condition in solving an inverse problem may rise from the distribution of the prior data, physical properties of the problem or numerical limitations. This modification was done by defining an acceptance rule for the models generated inside promising cells which are located close to the complex boundary. These cells might be cut by the boundary lines and a very small volume of the multi dimensional search space may still covered by the selected cell. This reduces the chance of populating the promising cell with new models. They proposed excluding the promising cell if the number of rejected models outside the complex boundary exceeds a threshold. The excluded cell is then replaced with a new cell having the best misfit. Details of this implementation can be found in Wathelet [2008]. The modified version of the NA has been applied to a number of problems, including the work of Nguyen et al [2009] for wave inversion. However comparison of the performance of the modified NA with the original NA and other stochastic methods are currently under investigation.

Arwini and Stephen [2010] proposed a workflow to increase NA convergence speed. The idea is to use a proxy-driven gradient by estimating the relationship between changing uncertain parameters and the obtained misfit values. In the original NA algorithm, misfit was assumed to be constant within each Voronoi cell. The improved version proposed by Arwini and Stephen builds a linear probability inside each cell rather than assuming a constant misfit surface. This approach has resulted in an increase of convergence speed by factor of 2 to 3.

3.3.4 NA Applications

NA originally was applied to solve seismology problems. It has been applied to many problems, especially when there is a complex relationship between unknown parameters of the problem and the observed data. Marson-Pidgeon [2001] applied NA for seismic waveforms inversion. Cua [2005] used NA for finding the most probable magnitude and location estimates from the ground motion data, in an early seismic warning system. Other researchers have also used NA for waveform inversion including Yoshizawa and

Kennett [2002], Vallee and Bouchon [2004], Jansky et al. [2007], Kennett [2006] and Wathelet [2005].

Steffen [2006] solved the inverse problem of determining viscosity distribution in the earth's mantle, using the Neighbourhood Algorithm. Cerv et al. [2007] analyzed the deep structural uncertainty in seismoactive and volcanic areas given the geoelectrical induction data using NA. Recently Yao et al. [2008] used NA for inversion of seismometer data in order to determine the 3-D shear wave speed in the crust and upper mantle. Hadidi and Gucunski [2009] used NA for probabilistic inversion problems in pavement and geomechanical engineering.

3.3.5 Application of NA in Petroleum Engineering

Soon after the introduction of the Neighbourhood Algorithm, it was applied to petroleum engineering. One of the pioneering works in this area was by Christie et al. [2002] where they used NA for obtaining multiple history-matched models in the Teal South reservoir. The Teal South example will be used in chapter 4 to illustrate the application of stochastic agent-based optimization methods in a simple problem. Subbey et al. [2003] applied NA to the SPE-10 case study using a streamline simulator for speeding up the history matching.

In more challenging problems, Litvak et al. [2005] stated that the neighbourhood algorithm has been successfully applied for history matching several BP operated reservoirs in the Gulf of Mexico (GOM). Nicotra et al. [2005] also reported the application of NA to history matching the Rigel oil field operated by ENI. History matching the Rigel case involved perturbing 11 fault transmissibility modifiers. They showed how NA obtained better models with lower misfit values in comparison with the ones obtained via the manual history matching process. Valjak [2008] also used the Rigel field to study the performance of NA in automatic history matching frameworks. The neighbourhood algorithm has also been applied to history match an offshore gas field called Mistral with 7 wells and 6 years of production history [Rotondi et al. 2006].

Many works have reported using NA in joint history matching of production and 4-D seismic data. The papers published by Stephen and MacBeth [2006] amd Stephen et al.

[2007] give few examples of this application of NA. Sedighi and Stephen [2009] also used NA and a polynomial response surface proxy for speeding up the seismic history matching process.

Demyanov et al. [2004] coupled NA with a geostatistical framework and applied the methodology for history matching and uncertainty quantification in the PUNQ-S3 reservoir. In this framework, NA was used to perturb the properties of the geostatistical framework, instead of directly changing of reservoir model parameters. The PUNQ-S3 model will be used in chapter 5 to compare the performance of sampling algorithms for a high dimensional problem.

Arnold [2009] highlighted the importance of considering geological data in history matching and used the NA algorithm as the choice of sampling algorithm in the proposed history matching framework. Okano et al. [2006] proposed a new approach to quantify the uncertainty of relative permeability in coarse scale reservoir models using the neighbourhood algorithm. Suzuki [2007] used a modified version of NA coupled with a probability perturbation method [Caers, 2003] for history matching and uncertainty studies of complex geological structural. Oye and Roth [2003] applied NA for the problem of borehole microseismic data inversion in the Ekofisk oil field in the North Sea. Using NA within the automated framework, they identified different clusters of microseismic event locations in the field. Edris [2009] used NA as the sampling method for seismic history matching framework and studies the added value from incorporating 4-D seismic data in to the production data history matching.

NA has been mainly applied in low dimensional optimization problems. The number of unknowns are usually less than 25; for example 6 in Agostinetti et al. [2004], 10 and 11 in Vallee and Bouchon [2004], 21 in Resovsky and Trampet [2002] and 24 in Sambridge [1999 a].

3.3.6 Comparison between NA and other Optimization Methods

Sambridge in his first paper on NA [Sambridge 1999a], compared the performance of NA with a genetic algorithm (GA). He showed that NA performance is comparable or better than the GA. He used a test with an inversion of the receiver function for seismic

structure for his comparative study. He repeated both optimization runs for GA and NA with three different initial random seed values. Sambridge reported that NA in two of these tests got a better final misfit value. He suggested that GA tends to perform local searches, while NA performs a more global search. He also compared the sampling graphs of NA and GA, concluding that the broader sampling and the more global search of NA did not sacrifice the improvement of the fit of best model.

Agostinetti et al. [2004] compared the performance of NA with simulated annealing (SA) for mantle viscosity data inversion. He used two case studies for this comparative study. In a low dimensional problem with two unknowns, NA and SA both provided good results. However he states that as they moved to a higher number of unknowns (n=6), NA was more effective, being able to avoid the local minima and high misfit areas in the search space.

Suzuki [2007]'s work is also one of the few side-by-side comparisons of NA method with other stochastic optimization methods. Suzuki compared this technique with stochastic tree search (GNAT) which was proposed by Brin [1995], and concluded that the NA was more efficient than the GNAT algorithm, requiring less forward simulations to obtained history-matched models.

Erbas [2006] made an in-depth comparison between performance of NA and GA. First she studied a mathematical function with two variables which had one global minimum and two local ones. She concluded that in mapping the misfit surface, the genetic algorithm performed better and NA missed one of the local minima. She then applied these algorithms to a history matching problem. In her first test, she compared NA and GA using the SPE-10 model [Christie and Blunt, 2001] with 12 unknown parameters. For this case and the same number of function evaluations, GA was able to obtain slightly better models and also reduce the misfit more efficiently than NA. The second example in this comparative study was the IC fault model which has a very complicated misfit surface with local sharp minima and e unknowns [Tavassoli et al. 2004, Carter et al. 2006]. For this model, the neighbourhood algorithm obtained lower misfit values and also required fewer simulations in comparison with the GA. Another case study in Erbas's work was an upscaled version of the SPE-10 model with coarser vertical grids.

This model has 4 parameters for history matching, but in comparison with the other low-dimensional problem (IC-Fault), its misfit surface was smooth. For this case the GA performed better in terms of the minimum misfit. At the end Erbas compared the performance of GA and NA for a real field history matching problem with 21 unknowns. She performed a test where the two algorithms started their search from the same initial population. The GA reduced the misfit more effectively; however NA found a better-matched model with a lower misfit value.

Further to the comparison of the final misfit values for NA and GA (which are problem-dependent), Erbas also made a comparison between the sampling behavior of the algorithms. She pointed out that in most cases the GA had a wider sampling in the search space producing "cloud-like" ensembles, while NA produced local clusters of models. Sambridge's [1999, a] observations however indicate that for his particular problem, NA gave a more global sampling of the search space. It is clear that the final results of any optimization method and its performance in sampling the search space depend on its configuration and tuning parameters. Erbas and Sambridge used different versions of GA with different tuning parameters and this may explain the difference observed between the performance of NA and GA for these case studies.

The neighbourhood algorithm demonstrated a good performance in comparison with GA and other stochastic sampling algorithms such as simulated annealing and search trees. Based on these results, we have selected NA as the benchmark algorithm in this thesis for various comparisons with ant colony optimization and differential evolution.

3.4 Neighbourhood Bayes Algorithm (NAB)

After exploring the search space and generating the ensemble of models by different optimization (sampling) algorithms, the next step is to draw inferences from the completed ensemble. In a Bayesian appraisal framework, the analytical solution of posterior probability requires the integration of the likelihood function over the all possible values of the remaining parameters. In this thesis, the NA-Bayes (NAB) algorithm [Sambridge, 1999 b] is used for posterior inference. NAB requires that forward simulation has been performed for all of the models in ensemble and their

(mis)fit to observed data are known. This step has already been performed using various sampling algorithms – in our case ACO_R, DE and NA.

NAB is a Markov chain Monte Carlo (McMC) method which builds an approximation for the real posterior probability distribution (PPD) using a Gibbs sampler. In fact, this is an interpolation problem in a multidimensional space. McMC techniques are used to explore the posterior distribution by sampling this space. In general, each sampling performed by a McMC method requires evaluation of the objective function at the specific point.

NAB uses Voronoi cells to represent the model space and to interpolate the PPD of unknown points in the search space. Voronoi cells act as a surrogate approach to estimate and interpolate unknown misfits (PPD) values. Voronoi cells assume that the misfit value is constant over each Voronoi cell. This interpolation of the misfit surface relaxes the need for running a forward reservoir simulation for posterior sampling. To summarize the benefits of NAB, the following two points can be highlighted:

1. All models in the ensemble are used to infer the information and to evaluate the posterior probability of the ones contributing to the confidence prediction. Hence, models with different goodness of fit contribute differently to the uncertainty prediction.
2. There is no need to run forward reservoir simulations for all the models generated by the sampling algorithm, but only for the ones resampled by NAB. This helps to save computational resources.

The working mechanism of NAB is summarized in figure 7 for a two dimensional problem and can be described as follows:

1. The algorithm starts the first step from an arbitrary location (a model in the sampled ensemble) and performs a series of random walks along each parameter axis in turn.

2. For each axis (parameter), a conditional probability distribution function (PDF) is created for the full parameter range (for example the XX' or YY' cut line in figure 7). For this purpose the intersections of each cut line with Voronoi cells are determined. First the intersections of the cut line with the neighbor cells are specified and this process is repeated for remaining cells on both sides of the selected axis until we cover the full parameter range. For each interval in the cut line, the conditional PDF is determined. The probability is determined from the product of the PPD value and the width of the intersection.

3. Each walk is performed on the selected axis by generating a uniform random deviation from the conditional PPD along the axis. The proposed step (x_i^p) is accepted or rejected based on following equation:

$$r \leq \frac{P(x_i^p \mid x_{-i})}{P(x_i^{\max} \mid x_{-i})} \tag{16}$$

where $P(x_i^{\max} \mid x_{-i})$ is the maximum value of the conditional PDF along the selected axis. r is a second random variable between 0 and 1.

4. If the step is rejected then the process is repeated.

5. The Gibbs sampler continues by generating the next step and cycles through each parameter axis in turn. An iteration is completed when all dimensions have been cycled through once.

6. After many independent walks starting from different locations, the constructed conditional PDF is believed to be a good approximation to the true posterior distribution. This process can be visualized as several thousand scans of the PPD surface on the parameter axes.

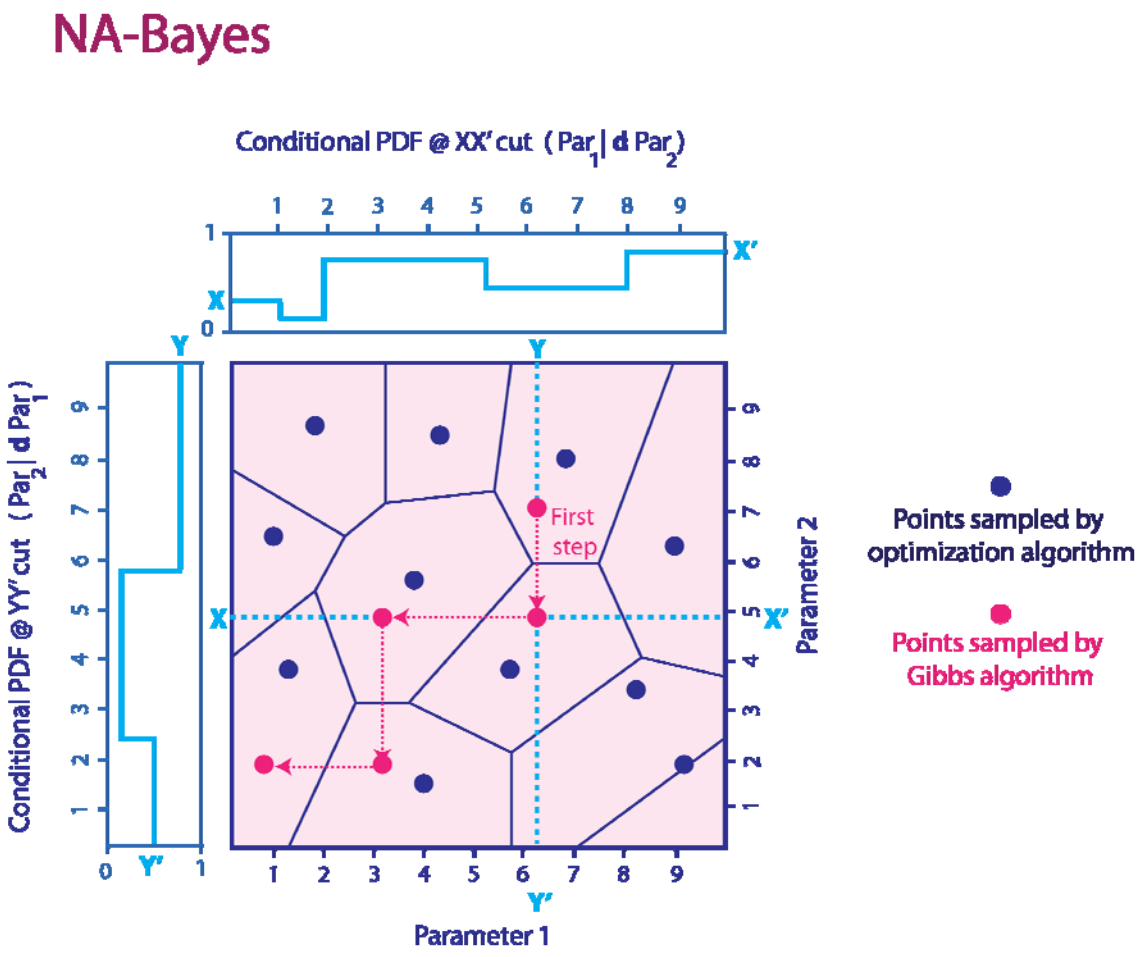

Figure 7: NA-Bayes algorithm

The accuracy of NAB depends on the complexity of the misfit space and the size of samples to represent this space. The shape and size (volume) of Voronoi cells are determined by distribution of the models using the optimization (sampling) algorithm. If there is a complex misfit surface (large change of misfit value within short distance) or limited number of samples (large Voronoi cells), then the NAB approach should be used carefully.

To investigate the computational efficiency of the NAB algorithm, a simple test was performed to understand the sensitivity of NAB to the number of dimensions and the ensemble size used for uncertainty quantification. In the first test, we chose a fixed ensemble size (1000) and tried 8 different numbers of variables (5, 10, 20, 30, 40, 50,

70, and 100). For the second test, 10 ensemble sizes were tested (100, 400, 700, 1000, 2000, 4000, 7000, 10000, 15000 and 25000) with a fixed dimension (5). Figures 8 and 9 present the results of these tests.

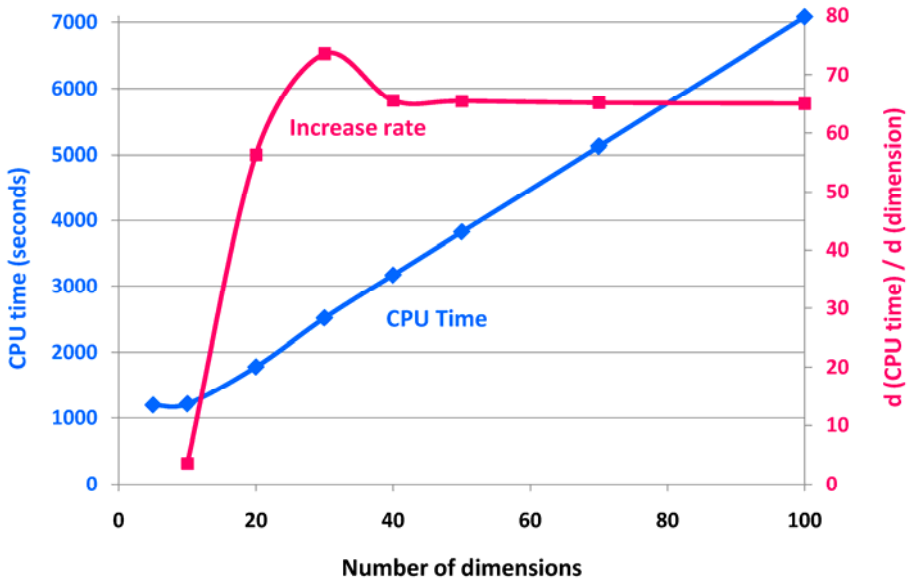

Figure 8: Number of dimensions vs. CPU time for NAB (ensemble size fixed at 1000)

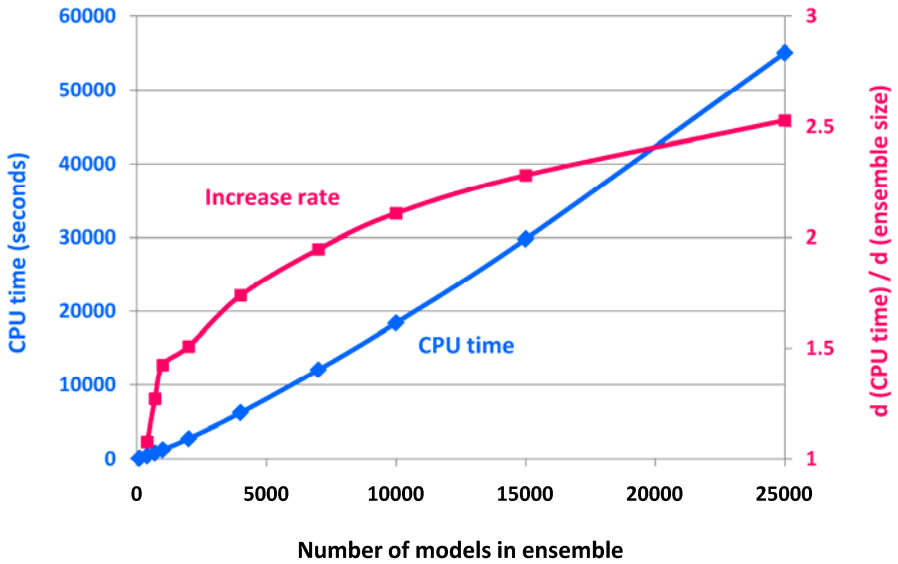

Figure 9: Number of models in ensemble vs. CPU time for NAB (dimensions fixed at 5)

These figures reveal interesting facts about the behavior of NAB by changing the dimensions in a given problem and the number of members in the ensemble submitted to NAB. Figure 8 shows that, as expected, the required CPU time by NAB increases as the dimensions of the problem increase. After a critical point (40 variables in this problem), the rate of the increase flattens to about 65 seconds per variable. This means that for each extra dimension, NAB spends 65 more seconds to finish the re-sampling process. In figure 9 we see the computational time required by NAB for different numbers of ensemble sizes. Again, as the number of models in the submitted ensemble increases, the CPU time increases. The rate of increase continues to grow as more models are added to the ensemble (at least up to 25000 models).

These tests show that for practical history matching problems, where dimensions are usually less than 100 and the ensemble size used for uncertainty quantification hardly exceeds 5000 models, the NAB algorithm provides a reasonable computation time to do the re-sampling process.

Chapter 4

> "Make everything as simple as possible, but not simpler"
>
> Albert Einstein

Teal South Reservoir Model: History Matching and Uncertainty Quantification

In chapter 3, we introduced three stochastic agent-based optimization algorithms for computer-assisted history matching framework. In this chapter ant colony optimization ACO_R), differential evolution (DE) and neighbourhood algorithm (NA) are applied to the history matching of the Teal South reservoir and uncertainty quantification of production estimates. The specific goals of this chapter are

1- To show the feasibility of using these algorithms to history match of a simple proof-of-concept reservoir
2- To study the effect of tuning parameters on the performance of ACO and DE and their stability to initial random seed of the algorithm
3- To investigate the uncertainty of predictions made by an ensemble of history-matched models
4- To understand the influence of available production data on the behavior of the algorithms during history matching and the uncertainty of the predictions

4.1 Teal South Reservoir

The Teal South reservoir is located in block 354, Eugene Island in the central Gulf of Mexico (figure 1). Block 354 is located approximately 257 kilometers southwest of New Orleans in a water depth of 85-90 meter [Roche et al. 1999]. This block originally was developed by Mobil Oil in the mid 80's. In 1987 Mobil left this block and Texaco acquired it in 1994 as a farm in from Shell. This block is currently being operated by Apache. Two wells were drilled in 1994. These wells identified several sand layers as potential targets for the next drilling phase. Eventually 17 wells were drilled. Most of the sands are over-pressured, highly laminated channels that generally extend in the north-south direction. The productive sands belong to Pilo-Pleistocene period [Fan et al. 2000].

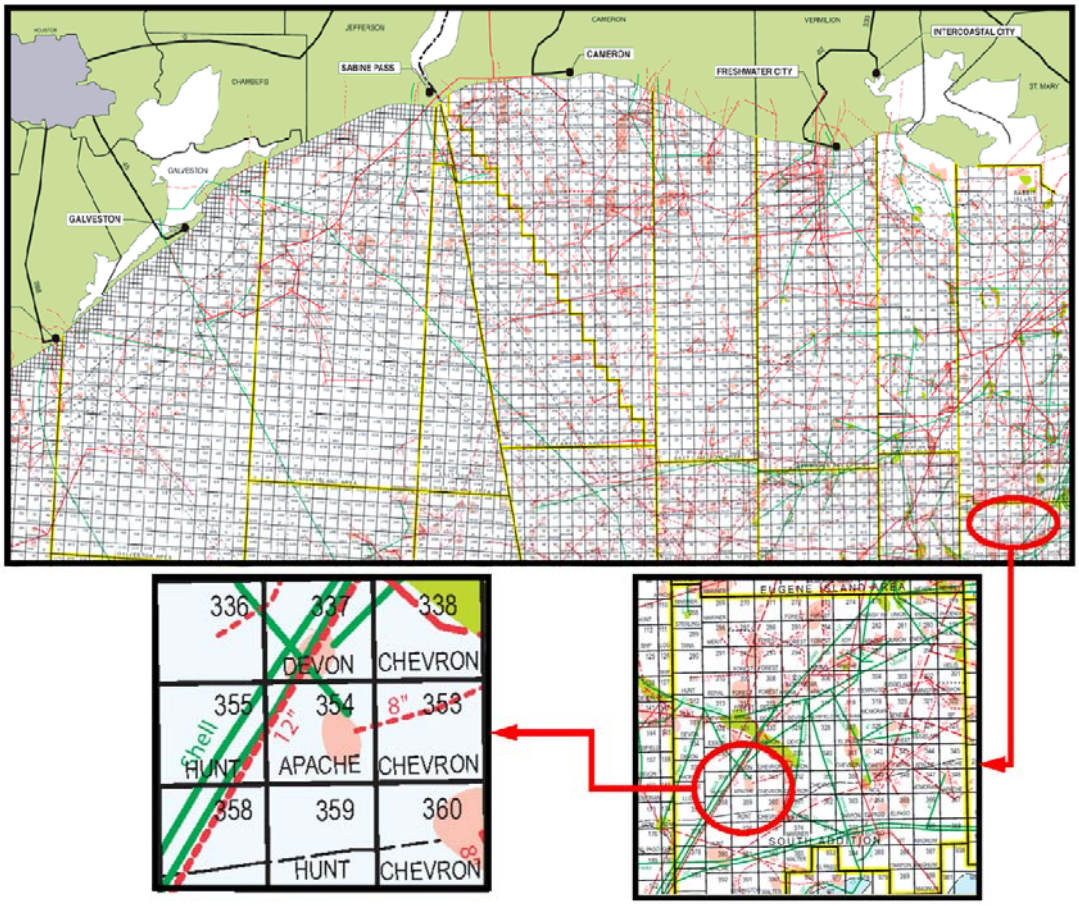

Figure 1: Teal South reservoir location in block 354, Eugene Island, Gulf of Mexico - map source from offshore magazine [2010]

The Teal South reservoir started production in late 1996. A single, horizontal well (D-10) penetrates the 4500 ft sand. In summer 1997, Texaco performed a data acquisition experiment using an ocean bottom cable (OBC) 4C*/3D seismic survey [Ebrom et al. 1998]. This survey acted as a base for a time-lapse study (4D) performed later by a consortium of 14 companies and 5 academic institutions under the management of the Houston-based Energy Research Clearing House (ERCH) [ERCH, 1998]. The reservoir sand is between two faults with sufficient throw to be visible at the seismic scale [MacBeth and Shams, 2006]. The structure map of this sand is available from the seismic survey.

A limited amount of reservoir data including some PVT data is provided for Teal South reservoir. There are only two pressure measurements: an initial pressure of 3096 psi and a measurement of 2458 psi after 570 days of production [Christie et al. 2002]. The production history of the reservoir consists of rates of oil, gas and water for 1247 days (figure 2). As we can see in this figure, the small reservoir volume and high flow rates result in rapid depletion [Purnell et al. 1999] and thus a reduction in production rates after 180 days.

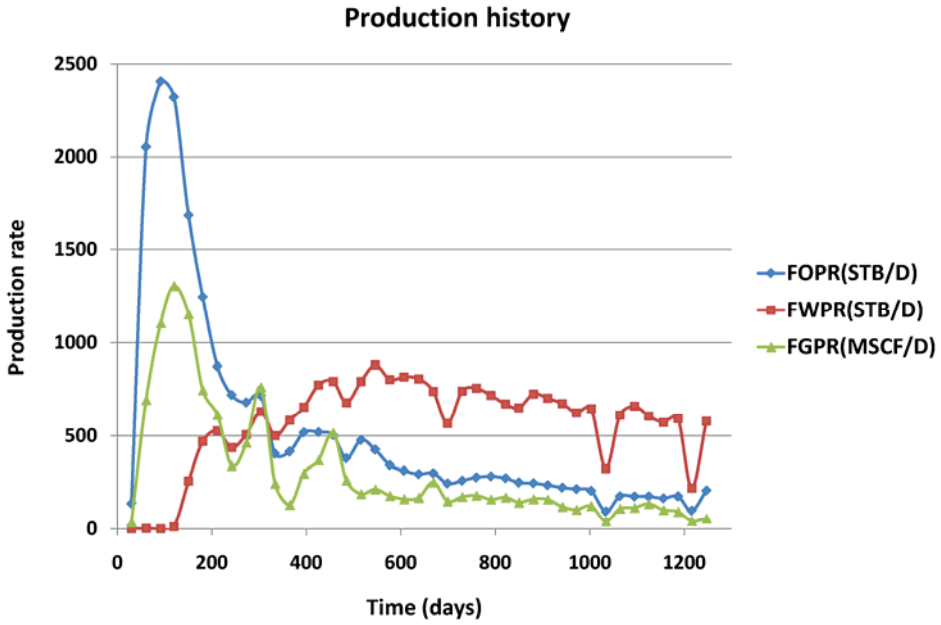

Figure 2: Production rates of oil (stb/d), gas (scf/d) and water (stb/d) from Teal South reservoir

* A 4 component (4C) seismic survey acquires the pressure and three orthogonal velocity components of the elastic wavefield

4.1.1 Parameterization of the Teal South Model

There are several studies on the history matching of the Teal South model with different geological parameterizations and unknown variables [Christie et al. 2006 - b, Pickup et al. 2008]. In this study we set up the simulation model on an 11×11×5 corner point grid (figure 3). There are five geological layers in the model with uniform properties. Porosity is assumed to be fixed at 28% through the reservoir. Key unknown parameters in history matching are horizontal permeability multipliers for each of these five layers (P1-P5), a single value for vertical to horizontal permeability ratio (P6), rock compressibility (P7) and aquifer strength (P8). Uncertain parameters for Teal South model and their prior range are shown in table 1.

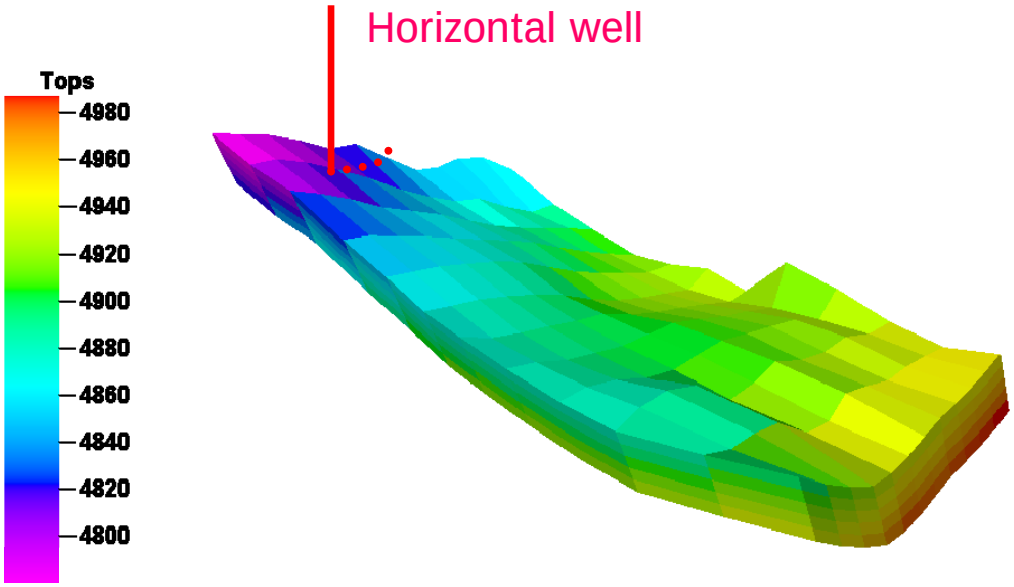

Figure 3: Teal South reservoir simulation model

Table 1: Uncertain parameters and prior ranges for Teal South model

Parameter	Unit	Prior range
k_h (for each layer)	mD	10 - 1000
k_v/k_h	-	10^{-4} - 10^{-1}
Rock compressibility	psi^{-1}	5×10^{-6} - 1×10^{-4}
Aquifer strength	MMSTB	10^{7} - 10^{9}

In this thesis, history matching was carried out by matching to the field oil production rate. We used different optimization techniques to minimize the following least square norm objective function in Teal South reservoir:

$$M = \sum_{n=1}^{N} \frac{(q_{obs}(t_i) - q_{sim}(t_i))^2}{2\sigma^2} \quad (1)$$

where N is the number of observations (oil rate measurements), q is the flow rate for observed and simulated data, and σ^2 is the variance of the observed data. The variance of oil rate measurements in the Teal South reservoir was assumed to be 100 stb/d.

4.2 History Matching

In this section we report the results of Ant Colony Optimization (ACO_R) and Differential Evolution (DE). These results include the best history match obtained, the sensitivity of ACO_R and DE algorithms to the starting random seed, the effect of tuning parameters on the performance of algorithms and a comparison of individual algorithms with the neighbourhood algorithm (NA) as the benchmark method.

The algorithms were coupled with the Eclipse™ reservoir simulator. Additional routines were used to prepare the input data for simulations based on the output of the optimization algorithm and post-process the simulation results to compute the misfit value. Both ant colony optimization and differential evolution algorithms were run in serial mode, which means simulations are sequentially performed on a single node of the cluster. Running 1000 simulations of the Teal South model took between 100 to 150 minutes. The neighbourhood algorithm benefits from a parallelization scheme and can be run on multiple nodes.

We used a random starting population in all of our simulations. However in real field applications there might be some information already available from previous studies about possible solutions for history matching. This information then can be used to narrow down the initial range for the parameters or act as an initial guide population for the optimization algorithm. In sections 2.1 and 2.2 the results of history matching using ant colony optimization (ACO_R) and differential evolution (DE) algorithms are presented.

4.2.1 Ant Colony Optimization (ACO_R)

This section presents the results of ant colony optimization for history matching of the Teal South reservoir. For having an initial idea about performance of the algorithm, several tests were performed using different combinations of tuning parameters. In each trial, all tuning parameters, except one, were kept constant and three values were tested for the varying parameter. Table 2 summarizes the tuning parameters and best misfit values obtained in each case using the ACO_R algorithm. In this table, m is the number of ants, k is the archive size, q is the search localization, ξ is the pheromone evaporation rate and M is the best misfit value. Each case has 1000 simulations.

Table 2: Different tuning parameters and best misfit values for initial test of ACO_R

Constant parameters	q = 0.001 m = 25 k = 50		ξ = 0.5 m = 25 k = 50		ξ = 0.5 q = 0.001 k = 50		ξ = 0.5 m = 25 q = 0.01	
	ξ	M	q	M	m	M	k	M
	0.1	16.0	0.1	14.7	10	14.6	50	16.8
	0.5	14.7	0.5	16.2	25	15.7	100	16.4
	0.9	16.4	0.9	15.4	50	14.5	200	16.5

4.2.1.1 Best History Matching Result

Among several trials, tuning parameters of the algorithm which resulted in the best match are presented in table 3. The best misfit value obtained is 14.5. Figure 4 shows the best history matched model in the ensemble which captures the observed oil production data points very well.

Table 3: Summary of parameters used in ACO_R for the best history matching result

Number of ants (m)	Archive size (k)	Search localization (q)	Pheromone evaporation (ξ)	Simulations
50	50	0.001	0.5	1000

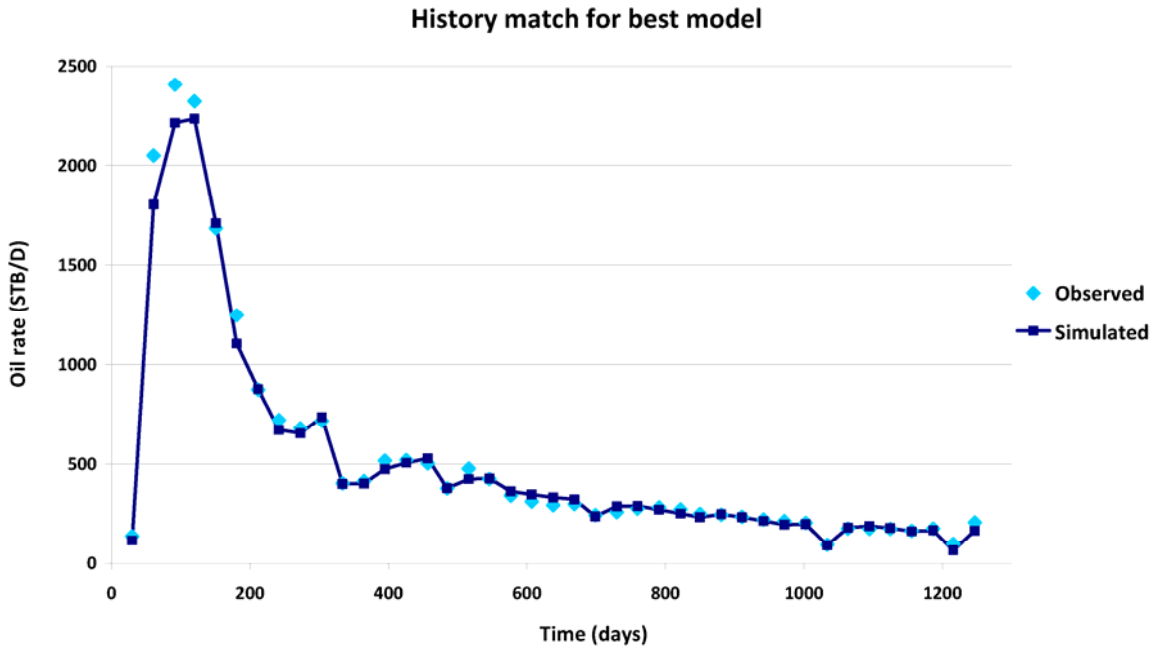

Figure 4: Simulated and observed oil rates for the best history matched model using the ACO_R algorithm

4.2.1.2 ACO_R Sensitivity to Initial Seed

Any stochastic sampling algorithm is sensitive to the initial random starting samples, and conclusions about the efficiency of any algorithm should be based on an average performance over a number of initial samples. To test the performance of the ACO_R algorithm we performed 10 runs of 500 simulations. These 10 runs all had the same run parameters, but had a different random set of initial samples. The parameters used in the ACO_R algorithm for this experiment are shown in table 4.

Table 4: Parameters used in ACO_R for 10 random runs

Number of ants (m)	Archive size (k)	Search localization (q)	Pheromone evaporation (ξ)	Simulations
10	50	0.01	0.5	500

For the Teal South, reservoir models with misfit values under 20 match the history well, with an average error of 1 standard deviation or less. As can be seen from figure 5

which shows the best misfit values obtained for each run, there were no failed runs among these 10 trials, e.g. models that were trapped in local minima had misfit values greater than 20. All the runs converged to regions of the search space with misfits smaller than 20. Although the number of simulations were considerably less than the ones used for finding the best model in the previous section (500 comparing to 1000), three of the models had minimum misfits between 15 and 16 which were close to the best model found in the previous section with a misfit of 14.5. Four models converged to regions with misfits between 16 and 18 and only three models had misfits between 19 and 20.

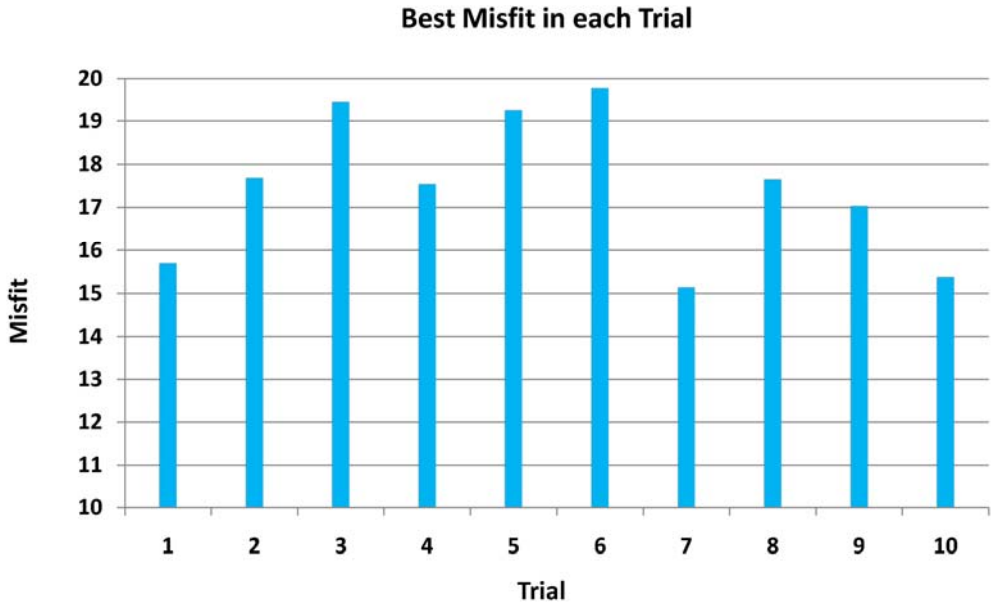

Figure 5: Best Misfit obtained for 10 random runs in ACO_R algorithm

4.2.1.3 Effect of Pheromone Evaporation Rate (ξ) and Search Localization (q) on the Performance of ACO_R

In ACO_R there are two parameters that control the behavior of the algorithm and its sampling performance in the search space. The search localization parameter (q) controls the preference to good models in ACO_R. This parameter adjusts the balance between diversification and intensification of the ACO_R algorithm. When q is small, the models which have higher ranks in the solution archive will be preferred. This is similar to choosing the best-so-far models. Then the search process will be focused around the best-so-far area of the search space. As the value for q is increased, the search will be

wider and we will increase the algorithm robustness. The pheromone evaporation rate (ξ) indicates how quickly ACO_R forgets the bad models. When ξ is large, the worst solutions will be forgotten faster (pheromone will evaporate faster) thus the chance of visiting previously explored regions will be higher; and the algorithm convergence rate will decrease.

To understand the effect of these tuning parameters, we performed two separate tests. In the first test we chose two search localization values (q=0.01 and q=0.5) with a fixed pheromone evaporation rate (ξ=0.5). The second test kept the search localization parameter fixed (q=0.5) and repeated the simulations for two different pheromone evaporation rates (ξ=0.5 and ξ=0.5). Each test was repeated 5 times to include the effect of initialization (random seed) on the results. Tables 5 and 6 summarize the tuning parameters used in these tests. Table 7 reports the best misfit values in each trial for different search localization and pheromone evaporation rates.

Table 5: Parameters used in ACO_R for testing the effect of q with ξ=0.5

Number of ants (m)	Archive size (k)	Search localization (q)	Simulations
25	25	0.01 and 0.5	1000

Table 6: Parameters used in ACO_R for testing the effect of ξ with q=0.5

Number of ants (m)	Archive size (k)	Pheromone evaporation (ξ)	Simulations
25	25	0.1 and 0.9	1000

Table 7: Best misfit values obtained for testing the effect of tuning parameters on performance of ACO_R algorithm (best misfit in each test displayed in bold)

Test type	Value	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5
Search localization (q)	0.01	**15.3**	17.4	16.6	17.1	16.4
	0.5	19.0	**15.0**	16.9	15.7	15.7
Pheromone evaporation (ξ)	0.1	16.2	18.0	15.8	**15.1**	16.7
	0.9	18.6	17.1	16.9	19.0	**16.0**

The first comparison between the effects of tuning parameters was focused on the sampling performance of the algorithm under different pheromone evaporation and search localization values. For this purpose, the trials that resulted in the best misfit values in the above tests are visualized using sampling history diagrams. These diagrams show each of the history matching parameters (P1-P8) and how sampling progresses during the history matching. Each figure has 8 tiles, showing 8 parameters of history matching. The horizontal axis shows the number of simulations and the vertical axis shows the scaled values of each parameter between 0 and 1.

Figure 6 shows the sampling history for the minimum misfit values obtained for two different search localization values. Examining this figure, we see that with smaller search localization values, the search space narrows down around the best available solutions, resulting in rapid convergence to optimal parameter regions. Near zero values, practically, means only best-so-far solutions is preferred by the ants. Higher q values (figure 6-right) result in a more global exploration of the parameter space by the ACO_R algorithm.

Figure 7 compares the sampling history figures for the minimum misfit models in the second test, in which we kept the search localization parameter fixed (q=0.5) and we studied the effect of pheromone evaporation rate on the performance of the ACO_R algorithm. We can see that a lower pheromone evaporation rate (ξ=0.1) results in a rapid convergence of the search towards good-fitting regions of the parameter space. However, the behavior of the algorithm using low values of pheromone evaporation is different from using low search localization values (figure 6). Although they both share a rapid progress towards final regions of interest for parameters, using low ξ values results in a competition between a *few* good and *distinct* areas of the search space at early stages of sampling. As the value of ξ increases, due to fast pheromone evaporation, the algorithm explores more regions of parameter space rather than concentrating in distinct areas.

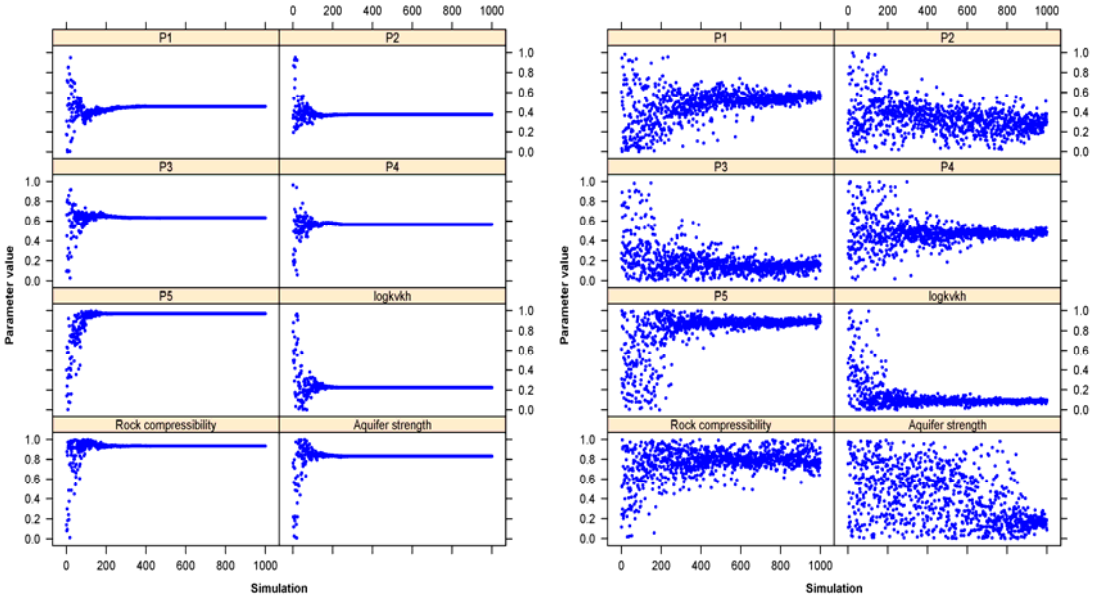

Figure 6: Sampling history for two search localization parameters (q=0.01 in left and q=0.5 in right) at a fixed pheromone evaporation rate

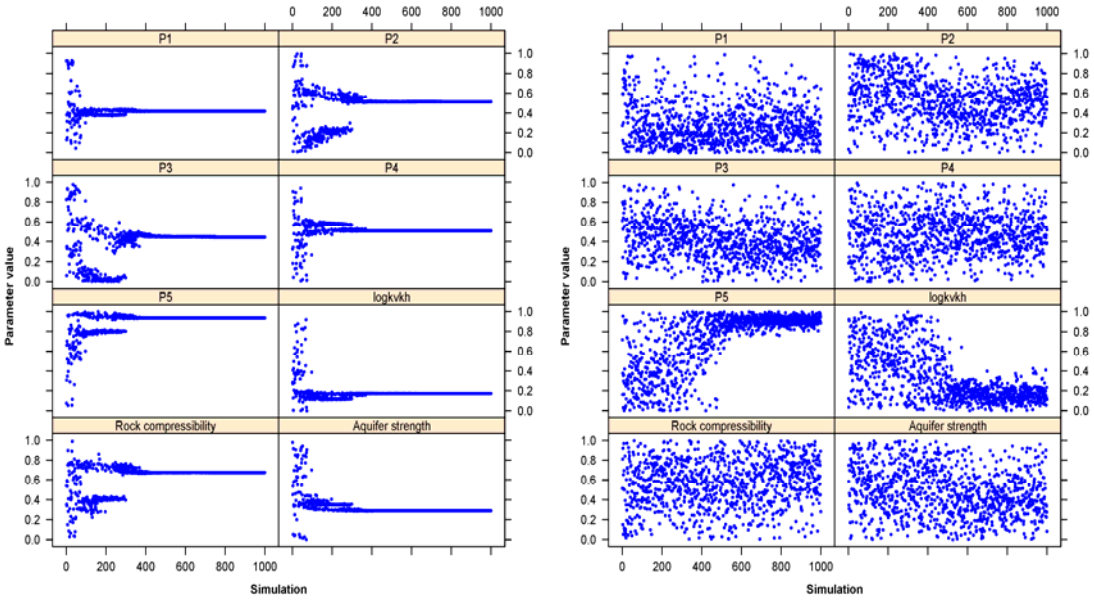

Figure 7: Sampling history for two pheromone evaporation rate (ξ =0.1 in left and ξ =0.5 in right) at a fixed search localization value

In addition to the visual comparison of the sampling history of the algorithm, we also investigated the effect of search localization and pheromone evaporation on the misfit of

models based on the 5 reported trials. For this purpose, we used a boxplot visualization of the results (figure 8). The boxplot, also known as the box and whisker plot, was created by Tukey [1977]. It is a standardized exploratory graphical method to show the distribution of a dataset by displaying 5 descriptive statistics. A boxplot shows the minimum, first quartile, median, third quartile and maximum values of the dataset. The upper and lower boundaries (hinges) of the box represents the 75^{th} and 25^{th} percentile of the data. In other words 75% and 25% of the data fall below these percentiles. The distance between the upper and lower hinge is called the interquartile range, often abbreviated as IQR. A line in the boxplot indicates the "median" (or central most value) of the data. The median should not be confused with mean as it is the numeric value separating the higher half of a dataset from the lower half. It is obtained by arranging data from lowest to highest and the middle number in this set is the median value. The "whiskers" of the boxplot show the minimum and maximum of the data. In the presence of "outliers" in the dataset, whiskers extend to their maximum 1.5 times the IQR.

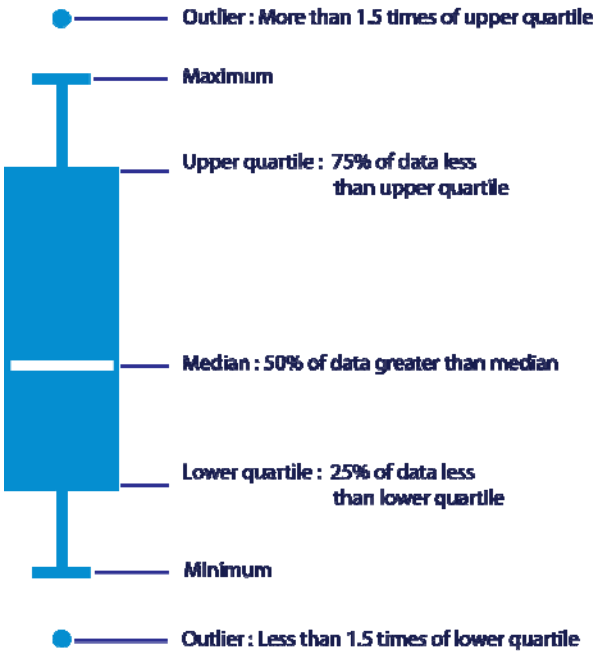

Figure 8: Boxplot diagram and its components

Figure 9 compares minimum misfit values for two search localization values in 5 independent runs and figure 10 shows the boxplots for the first objective function evaluation in which we get a misfit value less than 20.

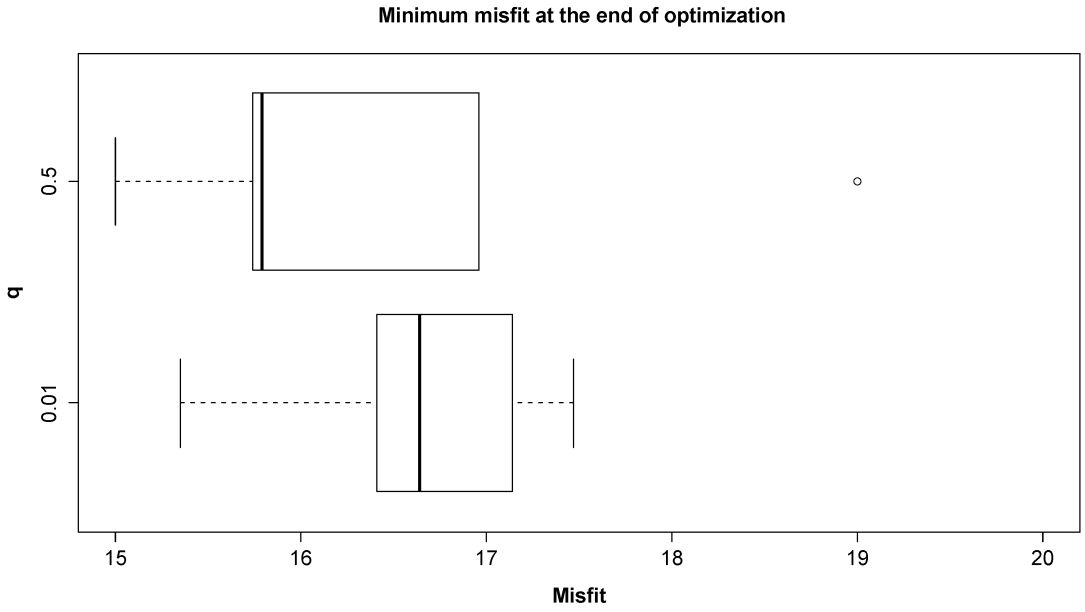

Figure 9: Minimum misfits obtained for two search localization parameters at a fixed pheromone evaporation rate

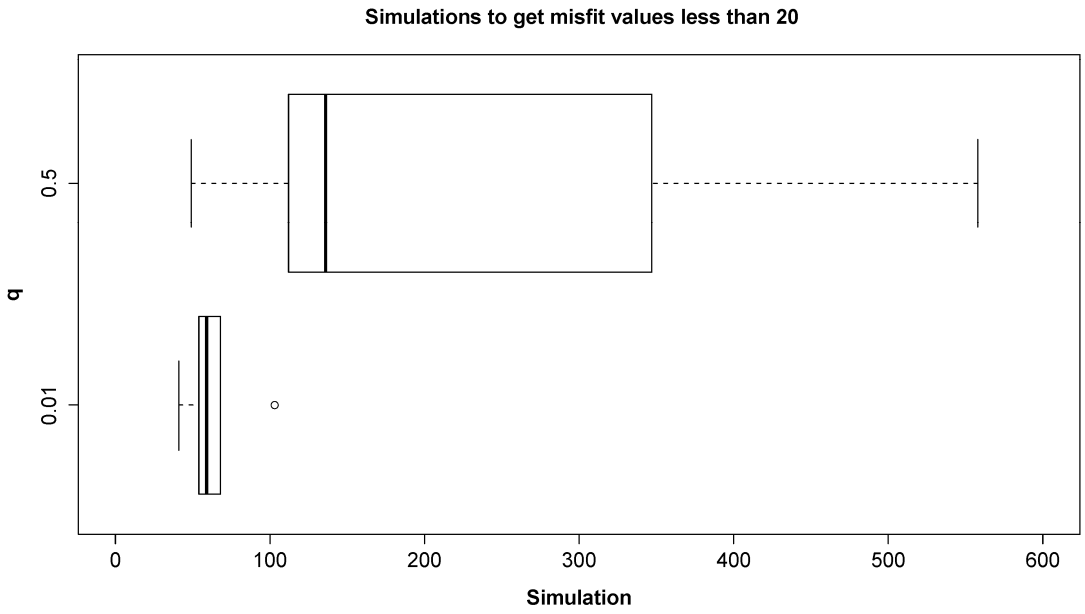

Figure 10: Number of simulation required to a get a misfit value less than 20 for two search localization parameters at a fixed pheromone evaporation rate

Figure 9 shows that for higher q values, on average, we get a lower final misfit value, however from figure 10, we see that it takes longer for the algorithm to find a model with a misfit value less than 20. The median of the boxplot for q=0.5 is 59 simulations, while this value for q=0.5 is 136 simulations.

Figure 11 compares boxplots of minimum misfit value for the second test, in which we keep the search localization parameter fixed (q=0.5) and study the effect of pheromone evaporation rate (ξ) on the performance of the ACO_R algorithm. Figure 12 compares the effect of two pheromone evaporation rates on the number of required simulation to obtain misfit values less than 20.

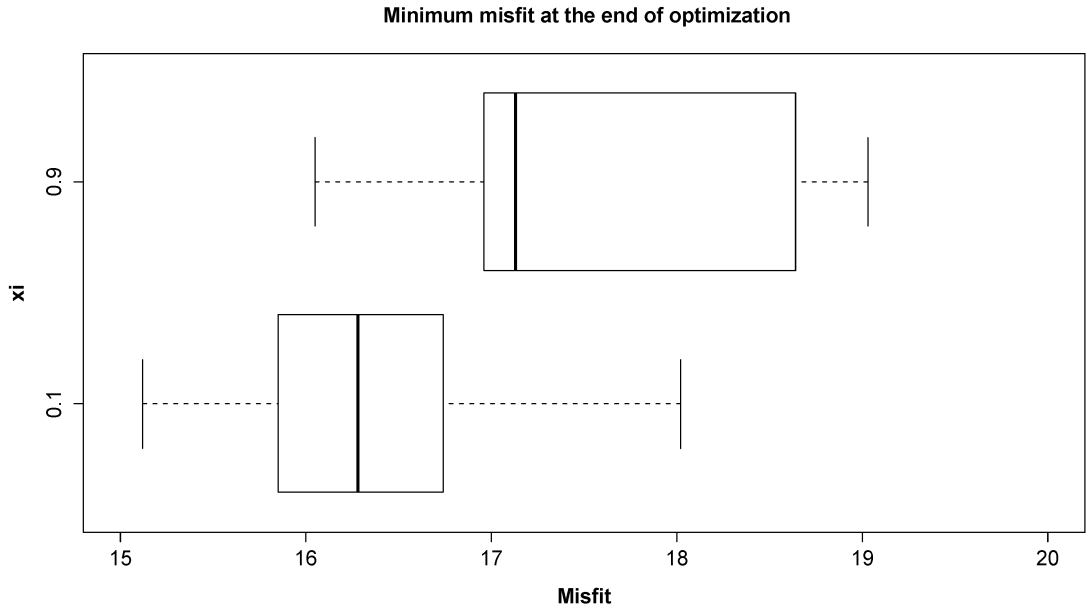

Figure 11: Minimum misfits obtained for two pheromone evaporation rates at a fixed search localization value

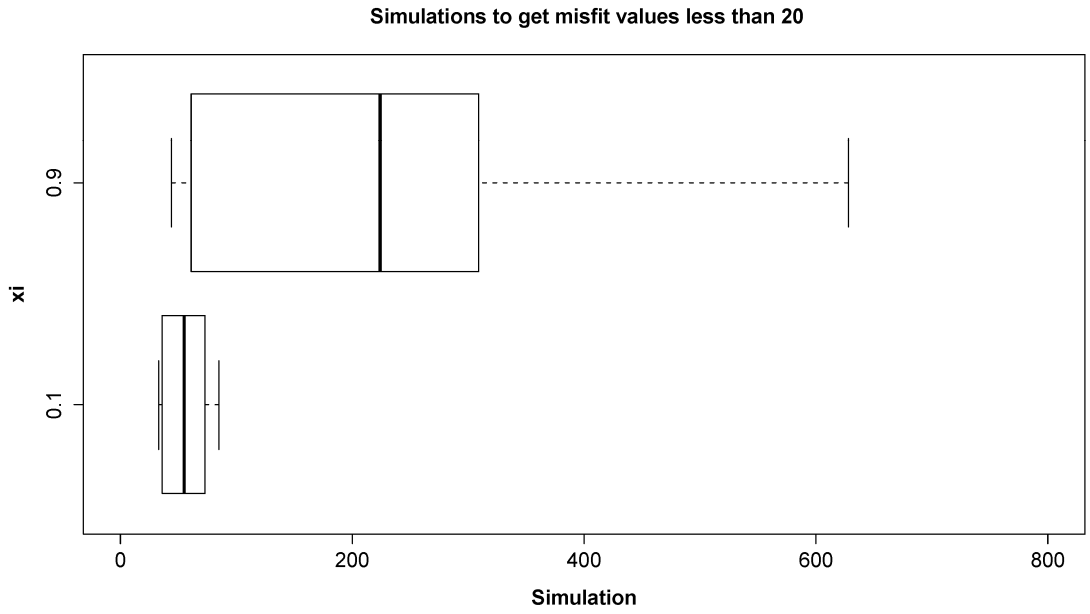

Figure 12: Number of simulation required to a get a misfit value less than 20 for two search localization parameters at a fixed pheromone evaporation rate

Figure 12 shows that with lower ξ values, the algorithm finds lower misfit values (M<20) faster. When ξ=0.1 the median of the boxplots was 55 simulations, while for ξ=0.5 the median value was 224 simulations. With lower evaporation rates, bad models are kept for a longer time in the solution archive and thus the algorithm concentrates on the regions with better misfit values.

4.2.1.4 Effect of Number of Ants (*m*) and Archive Size (*k*)

Remember from chapter 3 that the ACO_R algorithm starts the optimization by filling the solution archive (size k) with random solutions and then, at each iteration, m new ants are added to the solution archive. From the solution archive (now size k+m), the m worst solutions are removed to keep the archive sized fixed. Based on this update of the solution archive, the number of ants versus solutions in the archive must have an impact on performance of the algorithm. To investigate this issue, a test was performed with a fixed number of ants (m=25) and different archive sizes (k=25, 50, 100, 200). Table 8 presents the tuning parameters used in this test. Note that the results with m=k=25 is

taken from the test of search localization parameter given in table 7. Figure 13 summarizes the results of this test in boxplot form.

Table 8: Parameters used in ACO_R for testing the effect k and m (ant size is fixed m=25)

Search localization (q)	Pheromone evaporation (ξ)	Archive size (k)	Simulations
0.5	0.5	25, 50, 100 and 200	1000

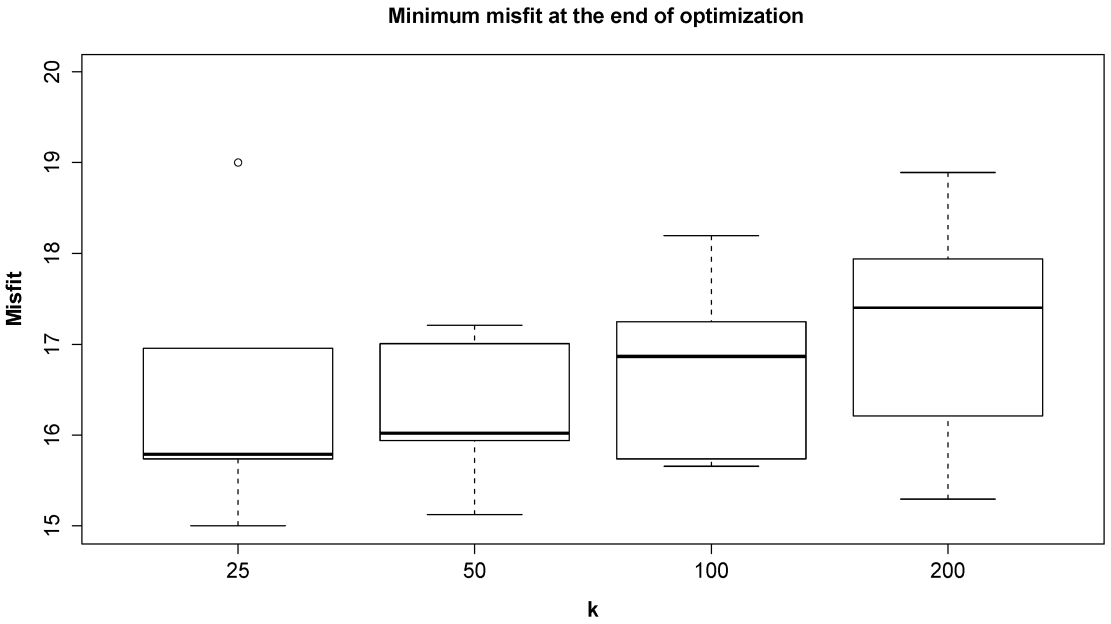

Figure 13: Minimum misfit obtained for different archive sizes and a fixed number of ants in ACO_R

As we can see in figure 13, for the cases where the ants number is same or close to the archive size, we obtain better results. With a fixed number of ants, larger solutions archives do not necessary provide better results. This is probably because for larger archive sizes, a small number of ants lose their effect on the archive. Furthermore, larger solution archive sizes, obviously, require more simulations at the initialization and are not helpful from the computational resource point of view.

4.2.1.5 Comparison of ACO_R with Neighbourhood Algorithm (NA)

In order to compare ACO_R and NA algorithms in terms of their misfit reduction efficiency, we attempted to set the NA parameters as close as possible to these used in the ACO_R run (table 9). The minimum misfit obtained for the ACO_R run with these tuning parameters was 14.7. For the NA case, we have tried 4 different setups with different algorithm parameters. Each of these four cases was repeated ten times in order to consider the effect of random initialization of NA. In each of these setups, there were 50 initial models and 25 cells were resampled at each iteration. The optimization continued for 38 iterations, which made the total number of models equal to 1000. The only difference between the different NA setups was the n_r value. Increasing n_r results in a wider exploration of parameter space. The effect of the parameter n_r in the NA algorithm can be considered to be similar to the parameter q in the ACO_R algorithm in which higher q values make the algorithm more explorationary. Summary of the parameters used in the NA runs for the Teal South reservoir (NA-TS) and the best misfit values in each setup and among 10 trials are presented in the table 10. The minimum misfit for the NA algorithm in these tests has been obtained in the NA-2 case with misfit of 14.8. We have chosen the NA-2 setup to compare the results with the ACO_R run.

Table 9: Parameters used for two different ACO_R runs

Case	m	k	q	ξ	Total simulations	Misfit
ACO_R-1	25	50	0.001	0.5	1000	14.7
ACO_R -2	25	50	0.1	0.5	1000	14.7

Table 10: Parameters used for four different NA runs

Case	n_{si}	n_s	n_r	Iterations	Total simulations	Misfit
NA-1	50	25	2	38	1000	18.7
NA-2	50	25	5	38	1000	14.8
NA-3	50	25	12	38	1000	15.0
NA-4	50	25	25	38	1000	15.1

Comparison of the best NA misfit (14.8) with the minimum misfit obtained by ACO_R with a similar setup (14.7) indicates that for this particular problem ACO_R obtains marginally better models than NA in terms of misfit value.

Figure 14 and 15 show the boxplot diagrams for the ACO_R and NA algorithms. Here we use boxplots to show the minimum, quartiles, median and maximum of the members in each generation. These figures show that ACO_R reduces the generational misfit values faster than NA, for the specific algorithm parameters used here.

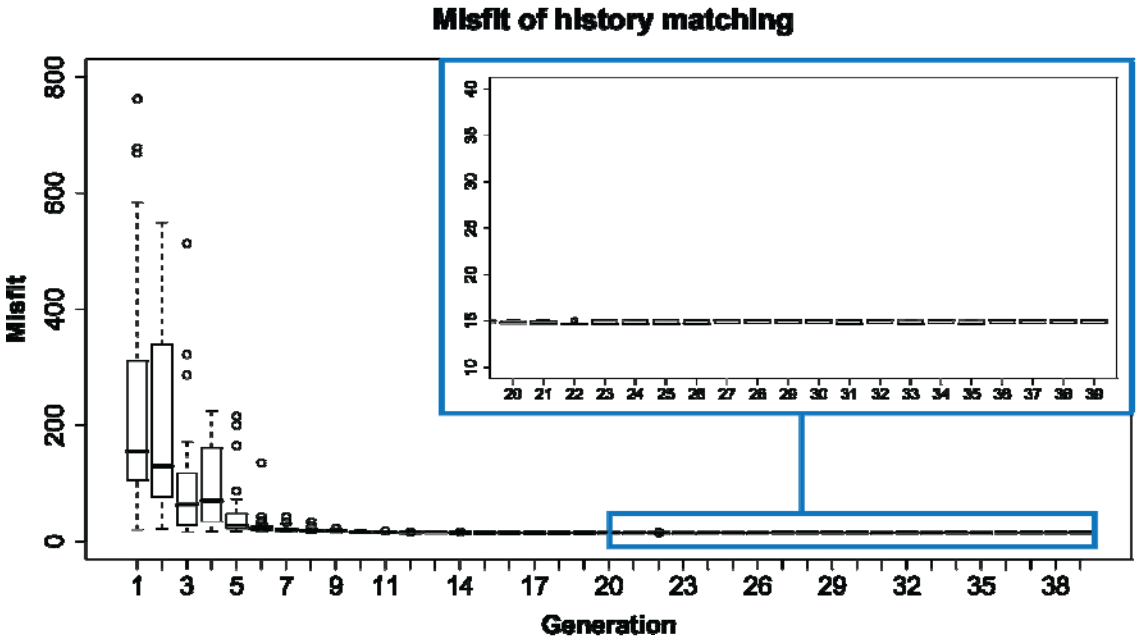

Figure 14: Boxplot for misfit values obtained by ACO_R

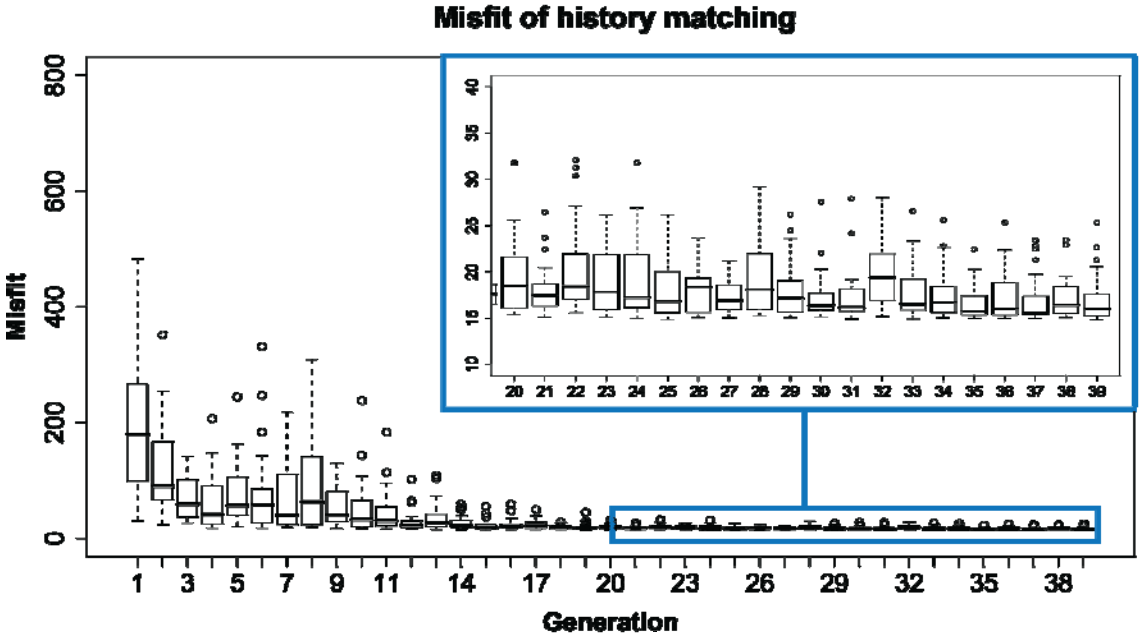

Figure 15: Boxplot for misfit values obtained by NA

Figures 16 to 18 show the sampling history of two ACO_R and one NA cases for each of the unknown parameters in the Teal South reservoir.

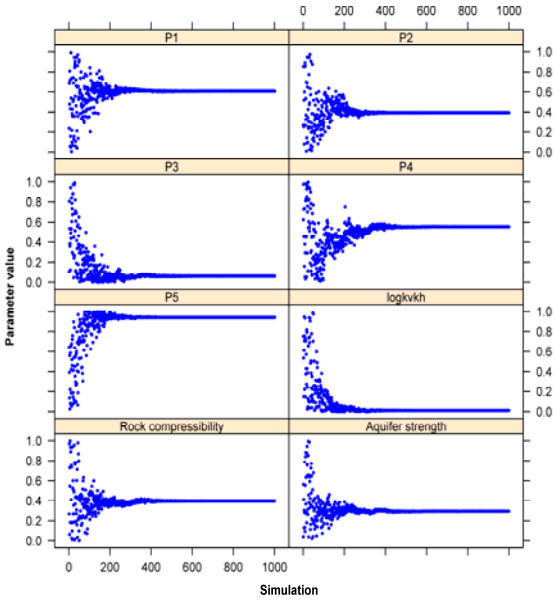

Figure 16: Sampling history for ACO_R-1

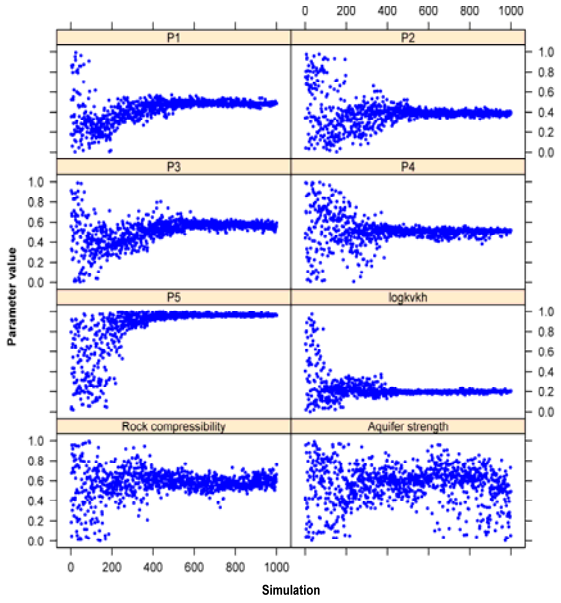

Figure 17: Sampling history for ACO_R-2

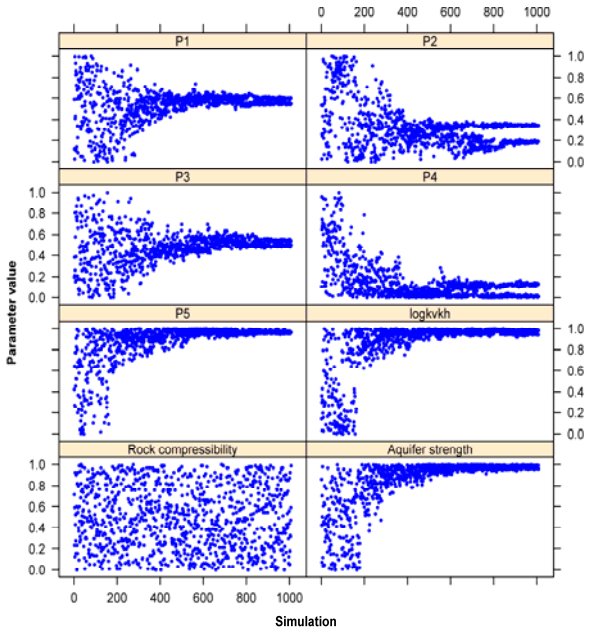

Figure 18: Sampling history for NA-2

From the above figures we can see that ACO_R-2 is able to maintain a larger population diversity due to its higher value for q. On the other hand ACO_R-1 has a faster

convergence to good fitting regions than ACO_R-2 due to its smaller q value, but this faster convergence comes at the expense of losing population diversity. Also we can see that NA-2 can simultaneously perform searches in two separate regions of the parameter space as it is progressing towards the end of history matching. Such performance is especially noticeable for permeability in layers 2 and 4 (P2 and P4) in figure 18. The geometry of Voronoi cells in NA allows the centre of sampling to change in different places, while simultaneously sampling from the best regions. Also as stated by Sambridge [1999 a], we should note the best fit model in each iteration is not necessarily obtained by sampling the previous best fit Voronoi cell, but may come from any other n_r cells. These two reasons help NA to simultaneously perform sampling in distinct areas of parameter space.

4.2.2 Differential Evolution

This section presents the results of history matching using the differential evolution (DE) algorithm and its comparison with the NA algorithm. For each strategy of differential evolution algorithm we have tried several history matching runs using different tuning parameters. In all runs the stopping criterion for DE is the maximum number of simulations which has been set to 1000. In all strategies, we have tried three different population sizes (N_p) (25, 50, and 100). Five values for the scaling factor (F) (0.3, 0.5, 0.9, 1.3 and 1.7) were used and for each of these scaling factors; we tried 3 different crossover (C_r) rates. This resulted in 45 different cases in each of the four strategies of differential evolution. Each case has also been repeated for three times to consider the effect of random seed value.

4.2.2.1 Strategy 1: DE-Rand

In this strategy, the base vector was chosen randomly and one weighted difference vector was added to it.

$$v_{i,G+1} = r_{1,G} + F(r_{2,G} - r_{3,G}) \qquad (2)$$

r_1, r_2 and r_3 were all chosen randomly. Table 11 gives the tuning parameters and best misfit values in each case for the DE-Rand strategy.

Table 11: DE-Rand cases and best misfit values

N_p=25			N_p=50			N_p=100		
F	C_r	Misfit	F	C_r	Misfit	F	C_r	Misfit
0.3	0.1	18.8	0.3	0.1	18.3	0.3	0.1	16.6
0.3	0.5	15.6	0.3	0.5	17.6	0.3	0.5	18.0
0.3	0.9	14.8	0.3	0.9	16.0	0.3	0.9	16.7
0.5	0.1	18.0	0.5	0.1	17.4	0.5	0.1	18.1
0.5	0.5	17.0	0.5	0.5	16.6	0.5	0.5	17.0
0.5	0.9	14.8	0.5	0.9	13.5	0.5	0.9	17.4
0.9	0.1	17.2	0.9	0.1	18.1	0.9	0.1	16.7
0.9	0.5	14.6	0.9	0.5	18.0	0.9	0.5	17.6
0.9	0.9	15.3	0.9	0.9	17.0	0.9	0.9	17.4
1.3	0.1	16.8	1.3	0.1	18.7	1.3	0.1	17.2
1.3	0.5	14.1	1.3	0.5	17.5	1.3	0.5	18.0
1.3	0.9	20.3	1.3	0.9	16.6	1.3	0.9	16.0
1.7	0.1	18.5	1.7	0.1	19.9	1.7	0.1	17.8
1.7	0.5	16.6	1.7	0.5	17.7	1.7	0.5	19.2
1.7	0.9	15.0	1.7	0.9	19.5	1.7	0.9	16.4

4.2.2.2 Strategy 2: DE-Best

This strategy works in a similar way to DE-Rand strategy, but instead of selecting the base vector randomly, the best vector with the lowest misfit value was selected as the base and a difference vector was added to it.

$$v_{i,G+1} = r_{Best,G} + F(r_{1,G} - r_{2,G}) \quad (3)$$

Here $r_{Best,G}$ is the best individual in each generation and r_2 and r_3 vectors are selected randomly from the current population. Table 12 reports the tuning parameters and best misfit values in each case for DE-Best strategy.

Table 12: DE-Best cases and best misfit values

N_p=25			N_p=50			N_p=100		
F	C_r	Misfit	F	C_r	Misfit	F	C_r	Misfit
0.3	0.1	18.5	0.3	0.1	16.4	0.3	0.1	19.7
0.3	0.5	14.5	0.3	0.5	15.2	0.3	0.5	16.5
0.3	0.9	16.0	0.3	0.9	16.6	0.3	0.9	16.0
0.5	0.1	16.1	0.5	0.1	16.6	0.5	0.1	16.4
0.5	0.5	16.5	0.5	0.5	15.0	0.5	0.5	15.6
0.5	0.9	16.7	0.5	0.9	18.7	0.5	0.9	15.9
0.9	0.1	17.4	0.9	0.1	15.5	0.9	0.1	17.5
0.9	0.5	15.1	0.9	0.5	16.5	0.9	0.5	16.0
0.9	0.9	17.8	0.9	0.9	14.3	0.9	0.9	16.4
1.3	0.1	18.8	1.3	0.1	19.5	1.3	0.1	16.7
1.3	0.5	15.9	1.3	0.5	14.8	1.3	0.5	18.3
1.3	0.9	14.7	1.3	0.9	14.6	1.3	0.9	16.6
1.7	0.1	17.7	1.7	0.1	18.3	1.7	0.1	18.6
1.7	0.5	15.7	1.7	0.5	17.5	1.7	0.5	16.5
1.7	0.9	14.8	1.7	0.9	16.7	1.7	0.9	16.4

4.2.2.3 Strategy 3: DE-Rand-to-Best

This scheme benefits from using both best the member of each generation and a randomly selected member. The Rand-to-Best strategy of differential evolution can be written as:

$$v_{i,G+1} = r_{1,G} + F(r_{Best,G} - r_{1,G}) + F(r_{2,G} - r_{3,G}) \quad (4)$$

Table 13 gives the tuning parameters and best misfit values in each case for the DE-Rand-to-Best strategy.

Table 13: DE-Rand-to-Best cases and best misfit values

N_p=25			N_p=50			N_p=100		
F	C_r	Misfit	F	C_r	Misfit	F	C_r	Misfit
0.3	0.1	16.3	0.3	0.1	18.5	0.3	0.1	17.1
0.3	0.5	15.8	0.3	0.5	17.3	0.3	0.5	16.4
0.3	0.9	18.3	0.3	0.9	18.3	0.3	0.9	15.7
0.5	0.1	17.3	0.5	0.1	16.0	0.5	0.1	19.8
0.5	0.5	16.5	0.5	0.5	19.0	0.5	0.5	15.6
0.5	0.9	15.7	0.5	0.9	15.3	0.5	0.9	15.8
0.9	0.1	16.7	0.9	0.1	17.0	0.9	0.1	17.5
0.9	0.5	18.1	0.9	0.5	15.0	0.9	0.5	15.2
0.9	0.9	15.3	0.9	0.9	15.5	0.9	0.9	15.6
1.3	0.1	17.3	1.3	0.1	16.8	1.3	0.1	17.9
1.3	0.5	16.1	1.3	0.5	15.0	1.3	0.5	17.9
1.3	0.9	19.5	1.3	0.9	17.0	1.3	0.9	15.2
1.7	0.1	19.6	1.7	0.1	17.6	1.7	0.1	18.6
1.7	0.5	19.4	1.7	0.5	15.0	1.7	0.5	18.2
1.7	0.9	16.3	1.7	0.9	16.6	1.7	0.9	18.9

4.2.2.4 Strategy 4: DE-Best-2

This strategy works in a similar way to DE-Best, but instead of one difference vector, two difference vectors were added to base vector. These two difference vectors were obtained from four randomly selected members in the population. The base vector is the best individual and two difference vectors were added to it. This is given by:

$$v_i = r_{best,G} + F(r_{1,G} - r_{2,G} + r_{3,G} - r_{4,G}) \quad (5)$$

where r_1, r_2, r_3 and r_4 vectors are selected randomly from current population. Table 14 reports the tuning parameters and best misfit values in each case for the DE-Best-2 strategy.

Table 14: DE-Best-2 cases and best misfit values

N_p=25			N_p=50			N_p=100		
F	C_r	Misfit	F	C_r	Misfit	F	C_r	Misfit
0.3	0.1	17.0	0.3	0.1	16.8	0.3	0.1	18.3
0.3	0.5	15.1	0.3	0.5	18.5	0.3	0.5	15.2
0.3	0.9	15.5	0.3	0.9	16.9	0.3	0.9	16.2
0.5	0.1	17.3	0.5	0.1	18.2	0.5	0.1	16.0
0.5	0.5	15.3	0.5	0.5	19.3	0.5	0.5	15.5
0.5	0.9	14.8	0.5	0.9	16.0	0.5	0.9	15.0
0.9	0.1	16.0	0.9	0.1	17.7	0.9	0.1	17.3
0.9	0.5	15.0	0.9	0.5	16.5	0.9	0.5	16.2
0.9	0.9	14.9	0.9	0.9	15.8	0.9	0.9	15.7
1.3	0.1	17.2	1.3	0.1	16.7	1.3	0.1	16.2
1.3	0.5	18.6	1.3	0.5	14.9	1.3	0.5	17.7
1.3	0.9	18.4	1.3	0.9	15.7	1.3	0.9	18.7
1.7	0.1	18.5	1.7	0.1	20.5	1.7	0.1	16.4
1.7	0.5	17.4	1.7	0.5	17.9	1.7	0.5	15.5
1.7	0.9	20.3	1.7	0.9	19.1	1.7	0.9	16.9

After running these simulations, we started a comparative study for four strategies of differential evolution. Comparisons were done in two separate areas: we studied the performance of the algorithm under different strategies and then we focused on the effect of tuning parameters

4.2.2.5 Comparison of Best History Matching Results

Figure 19 shows the best history matched model obtained in each strategy and its fit to oil production rate data in the Teal South reservoir. Table 15 gives the tuning parameters for each class of DE algorithm used in our study and the best misfit values.

Table 15: Tuning parameters used for each strategy in differential evolution and corresponding best misfit value

DE Strategy	N_p	F	C_r	Best Misfit
Rand	50	0.5	0.9	13.5
Best	50	0.9	0.9	14.3
Rand-to-Best	50	0.9	0.5	15.0
Best-2	25	0.5	0.9	14.8

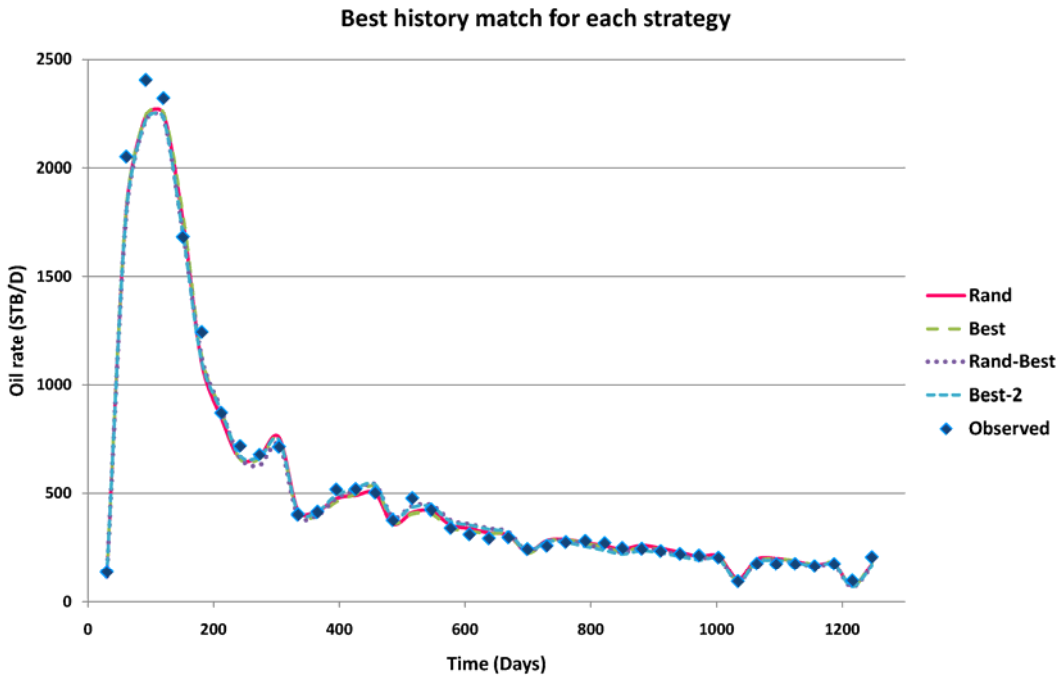

Figure 19: Best history matching results for oil production rate in each class of differential evolution algorithm

As one can see in figure 19, there is a small difference between all strategies of differential evolution in matching the oil production rate. This comparison shows all of the search schemes can get a good fit to observed oil rate data in this simple example. The minimum misfit and best history matching results are not the only target in any history matching study. This comparison does not reveal the difference between the performance of each DE search strategy used in our work. Any optimization method which is used for history matching should also have other characteristics like being robust to initial starting point of the search (not controlled by the random seed value) and effectively sampling the parameter space (influenced by structure and tuning parameters of the algorithm). In the following sections we will examine these issues.

4.2.2.6 Comparison of Sensitivity to Initial Starting Seed

Differential evolution is a stochastic optimization method which relies on generation of random numbers to guide its search. For an ideal stochastic method the final results should be independent of the initial seed used, however usually these optimization techniques are affected by their starting solution. In order to study the effect of initial starting conditions on the performance of differential evolution variants and their final results, a sensitivity study was performed. Ten independent runs, each with a random starting seed, were performed for each of the four search strategies used. Table 16 summarizes the parameters used for DE in this test. These parameters were used in all variants of differential evolution.

Table 16: Tuning parameters used for differential evolution in sensitivity test

Population size (N_p)	Scaling factor (F)	Crossover rate (C_r)	Simulations
25	0.3	0.5	500

Figure 20 shows the boxplots of the best objective function values obtained in each of 10 random runs and for all search strategies. Each boxplot shows the minimum, quartiles, median and maximum of results for runs performed in that strategy.

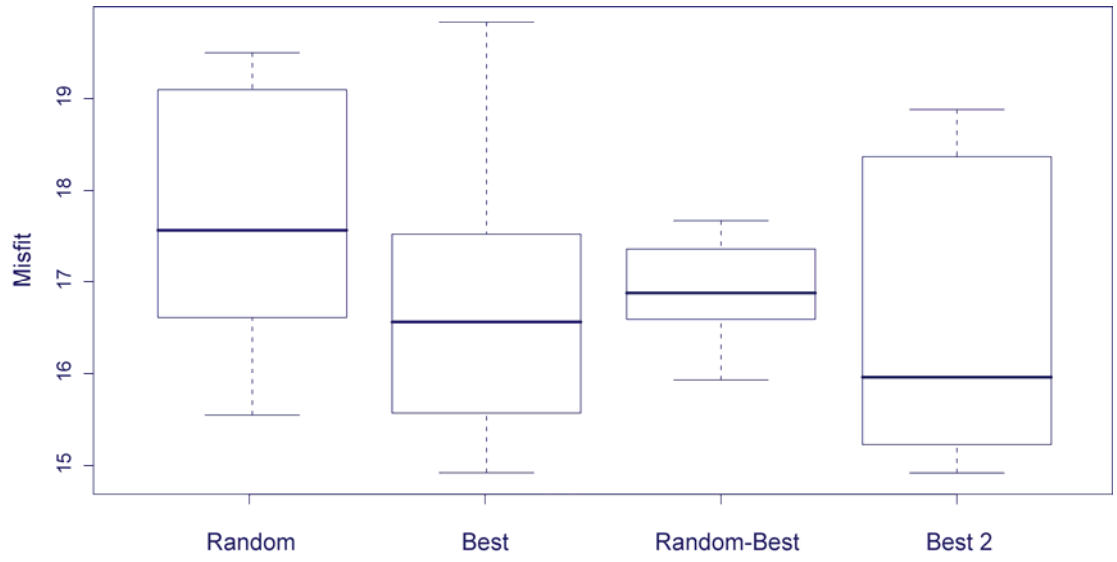

Figure 20: Box plots showing the results of sensitivity tests of differential evolution to initial starting seed

From figure 20 and by looking at the median values of the boxplots, we see that the performance of the DE-Best and DE-Best-2 strategies is better than the DE/Random strategy, considering the random seed effects. DE-Random-to-Best stands between the DE-Rand and DE-Best strategies. We also note the wide spread between quartiles of the boxplot for the DE-Best-2 strategy. This means although based on an average of 10 runs, it gets a lower misfit value, however its likely that for a single experiment we may not end up with a good match. The Rand-to-Best variant of differential evolution has the narrowest spread of quartiles in this problem. Also by considering the extreme values for the misfit boxplots, although running the history matching for only 500 fitness evaluations, in all of the 10 trials for the four strategies, the algorithm converges to good regions of the search space with misfit values less than 20.

4.2.2.7 Comparison of Convergence Speed in Different Strategies

In order to compare the convergence speeds of the different strategies in differential evolution for history matching, we have used the simulations presented in Table 16. In these experiments, simulations of each strategy have been repeated ten times to account for the effect of a random seed. In each strategy, the best misfits in ten trials were averaged and plotted in figure 21.

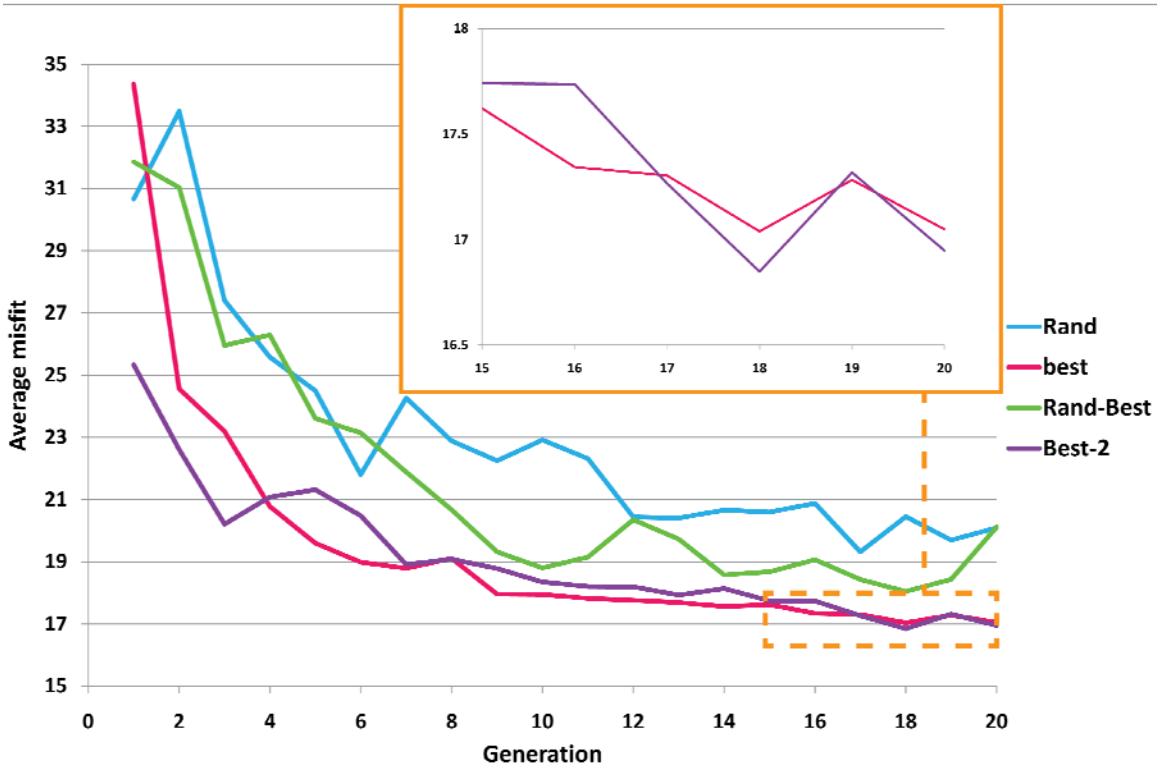

Figure 21: Comparison of convergence speeds for different strategies of differential evolution for history matching of Teal South model

Figure 21 shows that using the same tuning parameters in all strategies, DE-Rand has the slowest convergence. This behavior can be linked to the base vector selection mechanism in the DE-Rand strategy. DE-Rand-Best uses both tactics from the random and best strategy, and thus results in a convergence rate placed between the two. Both DE-Best and Best-2 strategies show a rapid convergence to good fitting regions of the parameter space. DE-Best-2 seems to speed up its convergence in generations after 15. We should not forget that these history matching runs had stopping criteria of 500 simulations, and therefore, in the next iterations, DE-Best-2 may have outperformed DE-Best strategy.

4.2.2.8 Comparison of Sampling Performance in Different Strategies

Further to the comparison of best match, seed-sensitivity and convergence speeds between different strategies of DE shown above, we may also compare the performance

of different variants of differential evolution in navigating the search space and how the possible solutions are identified. For this purpose we have performed a test for different strategies using the tuning parameters in Table 17.

Table 17: Tuning parameters used for differential evolution in sampling performance test

Population size (N_p)	Scaling factor (F)	Crossover rate (C_r)	Simulations
25	0.3	0.9	1000

Results from this test for DE-Rand, DE-Best, DE-Random-Best and DE-Best-2 strategies are shown in figures 22-25. In each figure, we show the progress of the differential evolution algorithm during history matching and the sampling of different parameters. Each graph shows 8 parameters with their scaled value between 0 and 1 plotted versus the iteration number.

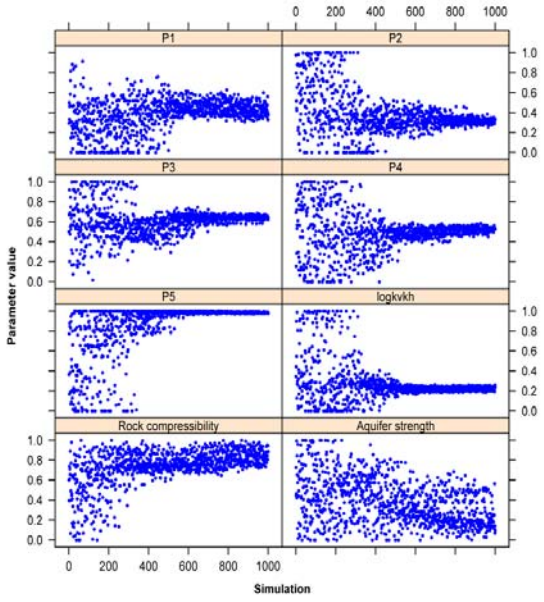

Figure 22: Sampling history for DE-Rand

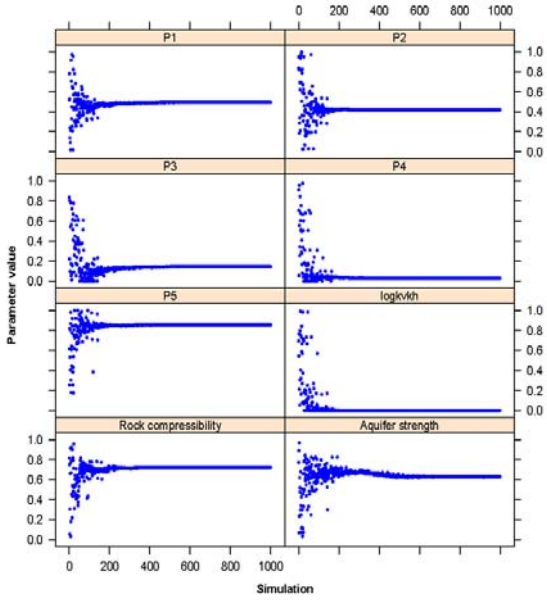

Figure 23: Sampling history DE-Best

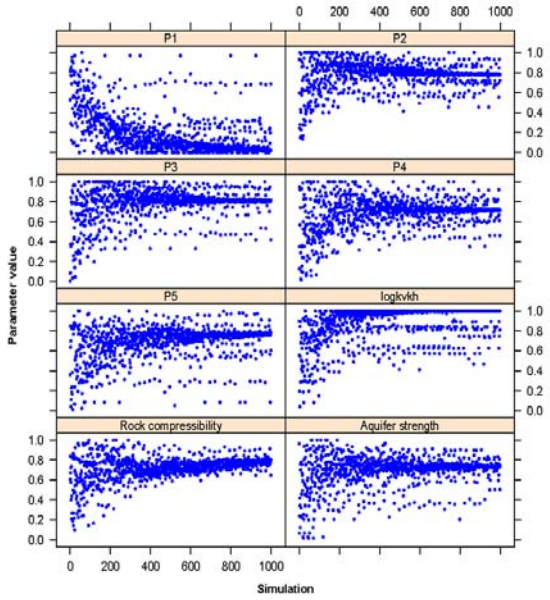

Figure 24: Sampling history for DE-Rand-to-Best

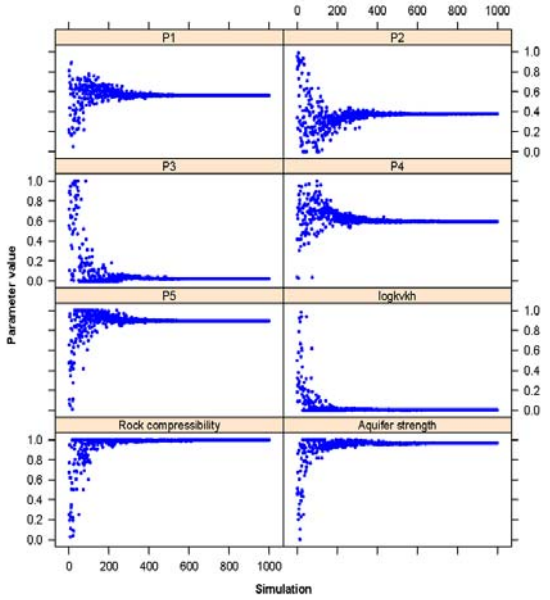

Figure 25: Sampling history DE-Best-2

Examining figures 22-25, we see that while we have a wide parameter space search for DE-Rand, in DE-Best and DE-Best-2 variants the eight parameters of reservoir model quickly converge to their final values. The performance of DE/Rand-to-Best in sampling the search space is between the DE-Rand and the DE-Best strategies. Another observation in figures 22-25 are the different final values for parameters of reservoir model. This behavior of the optimization algorithm arises from the inverse nature of the history matching problem, where multiple models with different simulation model parameters can yield a similar quality of history match.

4.2.2.9 Effect of Control Parameters on Performance of Algorithm

After comparing various strategies of differential evolution, in this section we investigate the effect of tuning parameters on performance of this algorithm. For this purpose, we have selected the DE-Rand strategy. We considered three cases: DE-1 and DE-2 have the same crossover value, but different scaling factors. DE-3 has the same scaling factor as DE-1, but with a different crossover rate. A summary of tuning parameters used for these three cases is given in Table 18.

Table 18: Summary of parameters used for checking the effect of tuning parameters on the performance of differential evolution

Case	Population size (N_p)	Scaling factor (F)	Crossover rate (C_r)	Simulations
DE-1	25	0.3	0.9	1000
DE-2	25	1.7	0.9	1000
DE-3	25	0.3	0.1	1000

4.2.2.9.1 Effect of Scaling Factor (*F*)

As mentioned in chapter 3, the scaling factor (F) controls the perturbation introduced to members of the population. It has no upper limit, but values larger than 2 have been occasionally reported in the literature. As F increases, the perturbation introduced to the selected base vector increases. Figure 26 shows the average minimum misfit obtained in each generation of search for two scaling factor values (0.3 and 1.7). We see in this figure that large F values result in a slower convergence for the DE algorithm in this history matching problem. The recommendation here is to use scaling factors less than 1.

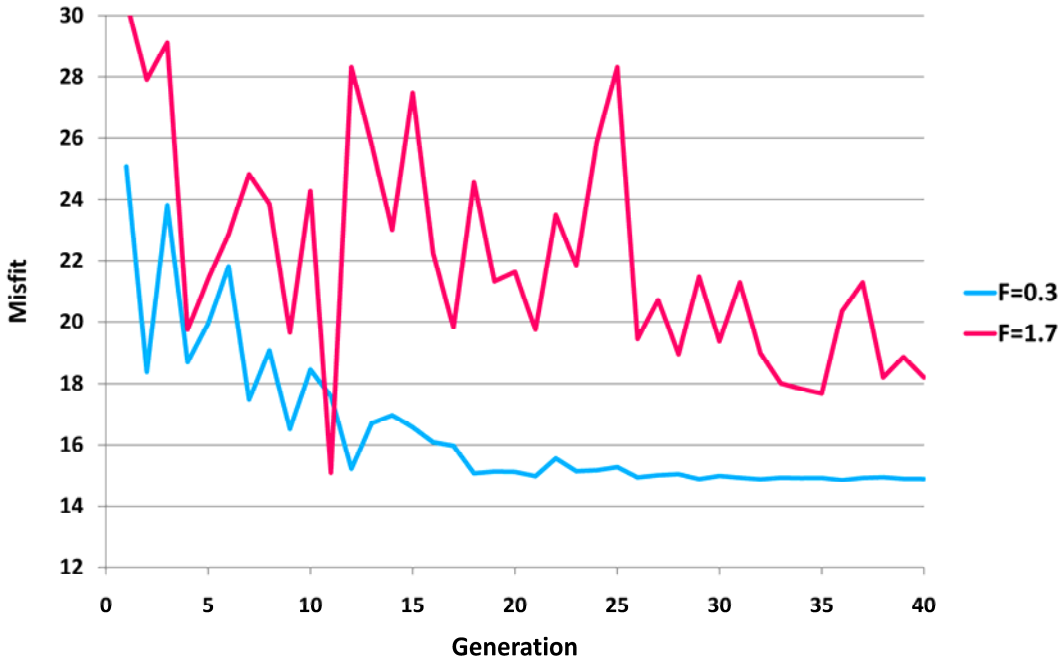

Figure 26: Minimum misfits in each generation for two scaling factor values

4.2.2.9.2 Effect of Crossover Value (C_r)

In differential evolution as the crossover value (C_r) increases, the convergence of differential evolution speeds up. In this section we have chosen two cases (DE-1 and DE-3) where they have same scaling factor, but different crossover rates. For each case we performed ten runs with the same algorithm tuning parameters, but starting from different initial points. We count the first generation in which we get a misfit value less than 20 and name this generation as Gen_{min}. Figure 27 shows the boxplots for DE-1 and DE-3 cases. Boxplots are used to show the minimum, quartiles, median and maximum of the members in each group.

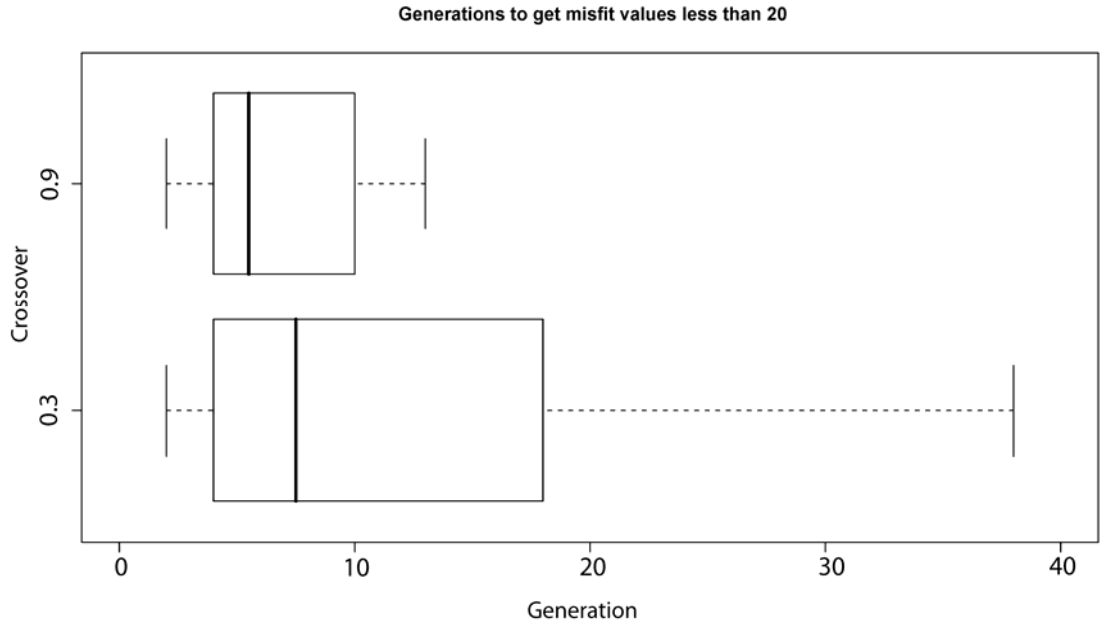

Figure 27: Boxplots for Gen_{min} in DE-1 (C_r=0.9) and DE-3 (C_r=0.1)

Figure 27 shows that for the higher crossover value we get faster convergence, and are able on average to generate a good history matched model by generation 6 or 125 runs of the reservoir simulator.

4.2.2.10 Comparison of Differential Evolution and Neighbourhood Algorithm

In a standard differential evolution algorithm the size of the population at the initial starting point and at other iterations are the same and equal to N_p. In order to compare differential evolution and neighbourhood algorithm in terms of the efficiency in misfit reduction, NA control parameters are adjusted to be similar to the tuning parameters used in the differential evolution algorithm. This parameter tuning includes fixing $n_s=n_{si}$ in order to keep population size at each iteration equal to the initial starting population in NA.

We describe two separate tests for the neighbourhood algorithm in this section. In the first test (A), the population size was 25 and in the second test (B) the population size was 50. Since the n_s/n_r ratio controls the behavior of the neighbourhood algorithm, in the case A we tried six n_s/n_r values and case B included seven different n_s/n_r ratios. One should note that increasing n_r in NA results in a wider exploration of the parameter space with $n_r=n_s$ being the most explorationary case. Each of these tests with different n_r values has been repeated for five times in order to consider the effect of random initialization of NA. The number of simulations for all of the cases equals to 1000. Due to different population sizes, we had 40 generations in the first test case and 20 in the second test case. Table 19 summarizes the setup conditions for tests A and B in the neighbourhood algorithm and the best misfits obtained in each case. The full results of these two cases are presented in figures 8 and 9.

Table 19: Best misfits obtained in two setups for NA with different population sizes and n_r values

Case	n_{si}	n_s	n_r	Generations	Simulations	Best Misfit
A	25	25	2,5,10,15,20,25	40	1000	14.8
B	50	50	5,10,25,25,30,40,50	20	1000	14.9

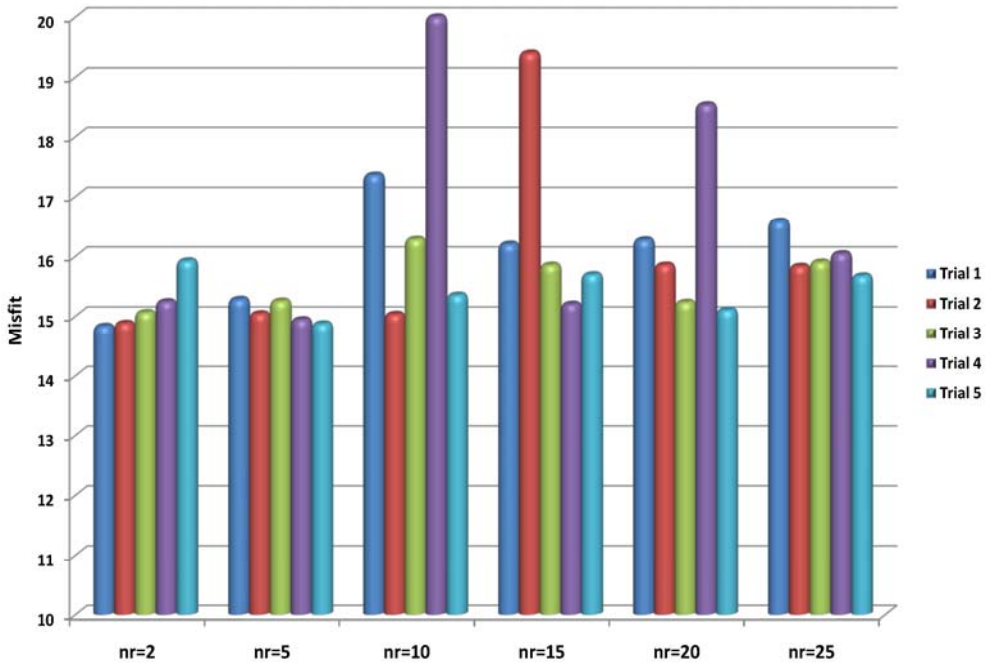

Figure 28: Best misfits obtained in each trial run with population size of 25 and for different n_r values

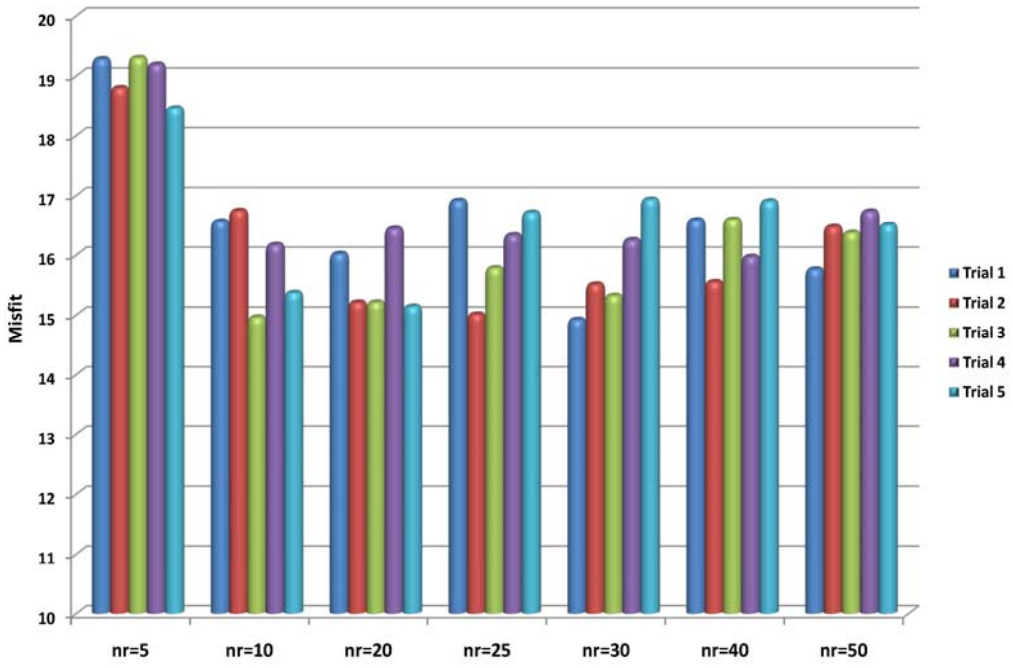

Figure 29: Best misfits obtained in each trial run with population size of 50 and for different n_r values

The minimum misfit obtained by NA in this test is 14.8 for a population size of 25 and 14.9 for a population size of 50. These values are comparable with previous runs of NA for the same case using a setup consistent with that used for the ACO_R algorithm. On the other hand, from Table 15, we recall the best misfits in each strategy of differential evolution: DE-Rand = 13.57, DE-Best = 14.36, DE-Rand-Best = 15.03 and DE-Best-2 = 14.88. Comparing the best misfit in DE and NA, we see a marginal difference between these two algorithms. While DE-Rand and DE-Best obtained a minimum misfit lower than NA, this algorithm obtained better misfit values in comparison with DE-Rand-Best and Best-2 (with equal number of agents per generation).

Figure 30 shows the number of generations required to achieve a misfit of less than 20 for NA and four strategies of DE. The random strategy of differential evolution, as expected, takes longer to find the first model with a misfit values less than 20. Also, DE-Best and DE-Best-2 performed well in finding low misfit models in the first few generations. We can see that both NA cases (NA-1 with n_r=5 and NA-2 with n_r=25) also quickly found a model with misfit less than 20. Having a higher n_r value makes NA-2, on average, slower in comparison with NA-1 case. On the other hand, DE is able to reduce the misfit to lower values than found with this set of NA runs.

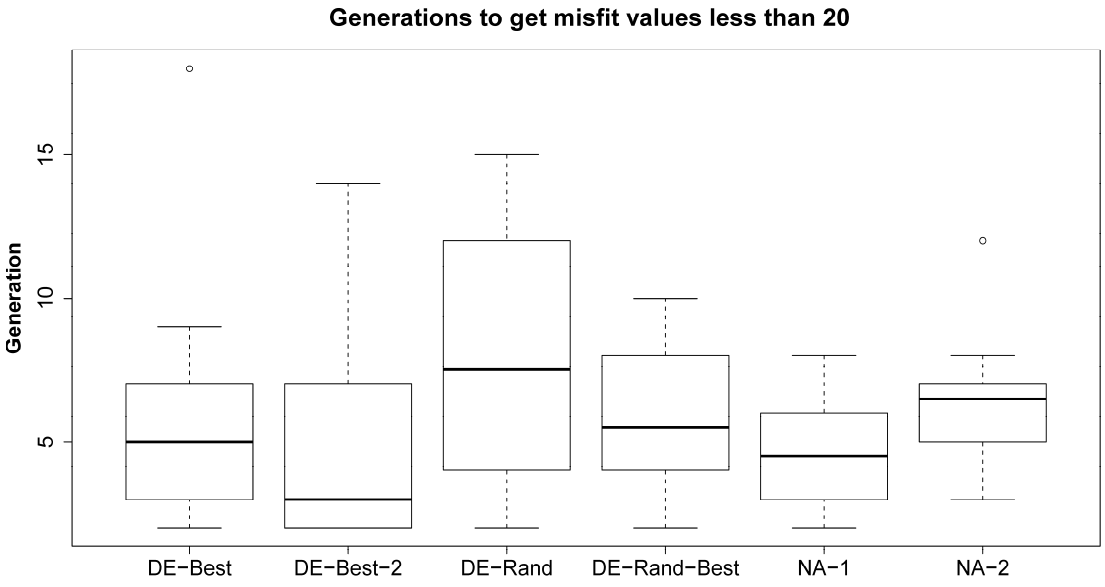

Figure 30: Boxplots for Gen_{min} in NA (n_r=5) and NA (n_r=25)

This test shows that it is still necessary to compare stochastic sampling algorithms from various viewpoints, and that the choice of comparison measure can affect which algorithm is portrayed as best. At the same time, we should also consider the complexity level of model. The marginal difference between minimum misfits for different strategies of DE and NA may be due to the simplicity of the Teal South reservoir model, the single match target and univariate objective function. In chapter 5, we provide a comparison between these algorithms when history matching a more complex reservoir with multiple match targets.

4.2.3 Value of Data in Uncertainty Quantification

In the history matching phase, the reservoir model is conditioned to the available production data from the reservoir. One of the key concerns for many researchers is the quality of the production data itself. There are several proposals to improve the quality of data using advanced data validation and reconciliation methods [Wising et al. 2009]. Although the quality of production data is one of the key focus areas in petroleum engineering, less attention has been paid to the importance of data itself. These days more high quality data can be obtained using digital oil field technologies [Lasrado 2009, Thomson 2008]; still the benefits and effect of additional information in field development studies remain an open research question. In this section we address the importance of data in history matching and its impact on the uncertainty of the predictions.

In order to assess the impact of historical data on the uncertainty of estimates, we consider two scenarios in this section. The number of oil rate measurements used for history matching is different in each case. In the first scenario, we have used 6 oil rate measurements during 181 days of production and in the second case we used 41 data measurement points corresponding to 1247 days of production. For the first scenario, we ran forward simulation after 181 days to obtain production data for 1247 days in the uncertainty quantification step.

We used ant colony optimization and differential evolution algorithms in these tests. The tuning parameters used for these algorithms are given in tables 20 and 21 respectively. The total number of simulations for each case in this study was 1380. We

started with an initial population of 30 and continued the optimization for 45 generations.

Table 20: Summary of parameters used in ACO_R

Number of ants (m)	Archive size (k)	Search locality (q)	Pheromone evaporation (ξ)
30	30	0.3	0.3

Table 21: Summary of parameters used in differential evolution

Population size (N_p)	Scaling factor (F)	Crossover rate (C_r)
30	0.5	0.5

The ensemble of 1380 models generated by ACO_R and DE were submitted to the NAB routine in order to determine the Bayesian credible intervals (P10-P50-P90) for the total oil production after 1247 days. Figure 31 shows the total recovery prediction from the Teal South reservoir in different scenarios studied in this section. The blue dashed line is the true total production after 1247 days which is computed using the real production data observed in the field.

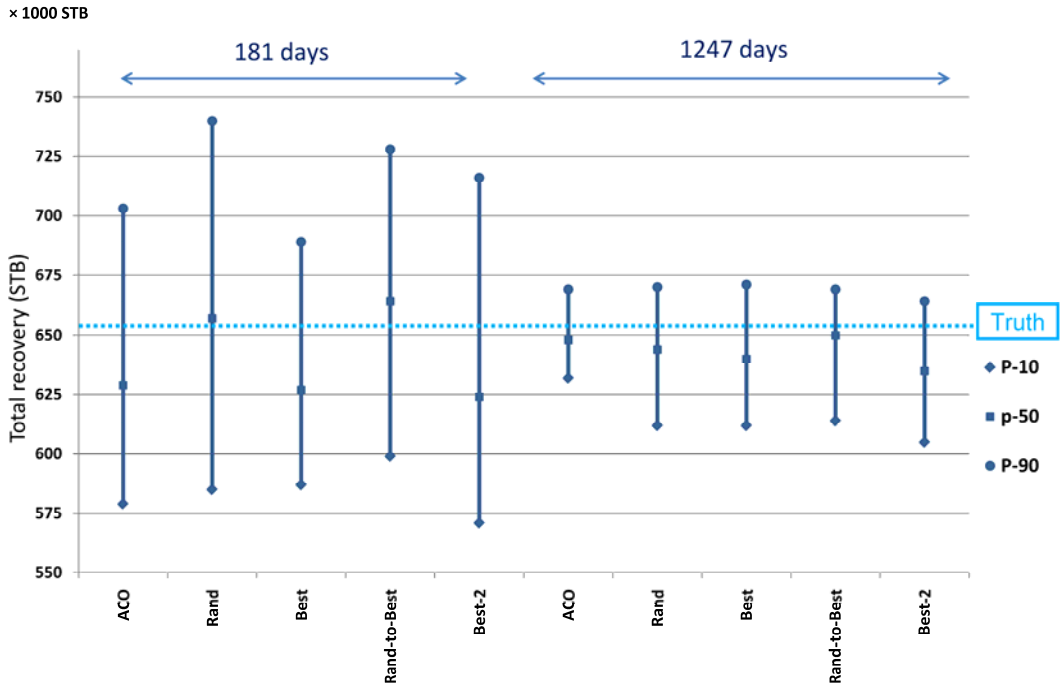

Figure 31: Uncertainty of the predictions made by ACO_R and DE in different scenarios

As seen in figure 31, we have narrower uncertainty ranges for the case of full history data. While in all cases P10-P90 ranges comfortably cover the truth production value, using more data points in history matching results in narrower uncertainty bands. In the first scenario, we have wide uncertainty estimates as the history matching was done using only 4 data points. In differential evolution, the DE-Rand strategy due to its random selection nature of the base vector, covered more regions of the search space and gave wider ranges for the prediction uncertainty. DE-Best and DE-Best-2 had a very fast convergence to optimal regions, without much chance to explore the whole parameter space; however they provide a reasonable uncertainty estimate for the case studied in this thesis. DE-Rand and DE-Rand-to-Best provided a P50 value closer to the truth value due to a wider sampling of the parameter space. We can also see that ACO_R usually provided reasonable uncertainty bands, especially when we used the full historical data. Another observation in figure 31 is the P50 value of the predictions which becomes closer to the truth total production as we introduce more data to the evolutionary optimization algorithms.

To summarize, in this section we studied the value of information in agent-based optimization for uncertainty quantification. We considered two different scenarios where the number of observation data points used in the optimization process were varied. Having additional information in history matching narrowed down the uncertainty intervals in predictions. This extra information also results in more accurate predictions for future performance of petroleum fields.

4.4 Concluding Remarks

This chapter introduced the application of ant colony optimization and differential evolution algorithms to the history matching of a simple reservoir simulation model. Teal South is a real reservoir with 8 parameters, a single well and an univariate objective function. The simplicity of Teal South allowed us to run multiple simulations in a reasonable time in order to understand various aspects of the performance measures of the algorithms.

In history matching, both proposed algorithms showed a better performance in comparison with the neighbourhood algorithm as the benchmark method. This

improvement comes in both minimum misfit values in history matching and convergence speed.

The population-based algorithms applied for history matching in this chapter offer more flexibility in their tuning. The neighbourhood algorithm only uses one tuning parameter (n_r or its ratio to n_s); however DE and ACO_R benefit from two tuning parameters. This gives an opportunity for the user to obtain the desired behavior in history matching. If the reservoir engineering studies are limited by computational resources, the algorithms can be set to converge faster. Otherwise we can allow these algorithms to have a more global exploration of parameter space and therefore more history-matched models.

In the uncertainty quantification step, the results of this chapter confirmed that all methods are able to provide an uncertainty envelope which covers the range of total production. The value of data used for history matching has also been studied and it is concluded that using more data helps to reduce the uncertainty of predictions.

In this chapter, we have developed a solid understanding about the performance of ant colony optimization and differential evolution and their tuning parameters on a simple model. In the next chapter, a more complex model with a larger number of unknowns will be considered to study the behavior of these algorithms in history matching and uncertainty quantification and to compare their efficiency in exploration/exploitation of a high-dimensional parameter space.

Chapter 5

> "As complexity rises, precise statements lose meaning and meaningful statements lose precision"
>
> Lotfi Zadeh

Comparative Study of Algorithms: PUNQ-S3 Model

In chapter 4, we applied different optimization algorithms to history matching the Teal South reservoir model. This case study had a simple structure and a single producing well. History matching of this model was a low dimensional problem with eight parameters and a univariate objective function. In this chapter, we compare the ant colony, differential evolution and the Neighbourhood Algorithms for a more challenging case. The PUNQ-S3 reservoir is a synthetic benchmark problem used in the petroleum industry to test different history matching and uncertainty quantification techniques. The PUNQ-S3 model has a more complex geological structure than the Teal South model, which entails solving a high dimensional optimization problem. This model is fitted to multivariate production data coming from multiple wells.

5.1 PUNQ Project

The PUNQ project stands for Production forecasting with UNcertainty Quantification. It was a joint effort of 10 European companies, universities and research centers supported by the European Union as a part of JOULE III program. The objectives of this project can be divided into two main groups - industrial and scientific. From the industrial point of view, the objective was to develop a method which could quantify production forecast without bias considering the uncertainties rising from reservoir modeling, reservoir parameters and well observations. The main scientific goals of the PUNQ project were to overcome the following challenges: parameterization of the model, optimizing the noisy objective function within a high dimensional search space with many local minima, while trying to minimize the number of required simulations. Within this project, two synthetic reservoir models were considered: a simple model (PUNQ-S) based on a real field in the North Sea operated by Elf Exploration and Production; and a complex model designed as an analogue for a Brent-type field with special sedimentological and structural complexities. The simple model, itself, had three variants, identified by PUNQ-S1, S2 and S3. These variants differ in the way in which the porosity and permeability fields were generated. In this thesis, we consider the PUNQ-S3 reservoir simulation model to compare our proposed algorithms. Stochastic correlation between porosity and permeability and injection of random noise to static and dynamic well data makes this case different from the PUNQ-S1 and S2 models [Boss 1999].

5.1.1 Literature Review of PUNQ-S3

The PUNQ-S3 reservoir model eventually became a popular benchmark model to test and compare novel methods developed for history matching and uncertainty quantification. Many authors have published the results of their research using the PUNQ-S3 reservoir model. Soleng [1999] used a steady state genetic algorithm for history matching of this model. Manceau et al. [2001] presented an integrated method for history matching and uncertainty analysis based on Fast Fourier Transform-Moving Average (FFT-MA) and gradual deformation techniques. Mantica et al. [2002] coupled chaotic optimization with gradual deformation to history match the PUNQ-S3 model. Fenwick and Roggero [2003] studied the updating of the matched reservoir model with new well data and the impact of this update on uncertainty analysis using a gradual

deformation method. Demyanov [2003] applied the Neighbourhood Algorithm (NA) coupled with a geostatistical framework. Ensemble Kalman filter (EnKF) has also been used for history matching of the PUNQ-S3 model [Gu and Oliver, 2005, Gao et al. 2005, Lorentzen et al. 2005]. Gao et al. [2007] also applied two types of Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm to history match the model. They compared the performance of this method with steepest descent, gradual deformation and Limited Memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS) algorithms. They concluded that none of the implementations of SPSA perform better than LBFGS either in robustness or computational efficiency. Recently Bourgault [2008] proposed the decomposition of the spatial properties into a trend and a residual component and then adjusting the parameters of trend and performing a stochastic simulation for the residual part. He applied these techniques to history match the PUNQ-S3 reservoir.

5.1.2 PUNQ-S3 Model

The PUNQ-S3 reservoir simulation model, as described by Floris et al. [2001], is a 5-layer model with a top depth of 2430 meters. It has a dip angle of about 1.5 degrees and is bounded by a fault to the east and south and has a relatively strong aquifer on the north and west that provides a pressure support. Because of this pressure support, no injection wells have been drilled in this reservoir. There is also a small gas cap in the PUNQ-S3 reservoir model in layer 1 and in the center of the dome-shaped structure. No well were completed in this layer to avoid free gas production.

Six production wells are marked with black dots in figure 1. These wells are located near the initial gas-oil contact. Producers 1 (PRO-1), 4 (PRO-4) and 12 (PRO-12) are perforated in layers 4 and 5. Producer 5 (PRO-5) and 11 (PRO-11) have been completed in layers 3 and 4 and producer 15 (PRO-15) has been perforated only in layer 4. There are also positions for five infill wells (X1-X5) which are shown as white dots in figure 1. PRO-4 is completed near the aquifer and water breakthrough has been observed in the 7^{th} year. Free gas production starts in 4^{th} and 5^{th} year in PRO-1 and PRO-4.

The PUNQ-S3 model has 19×28×5 grid blocks, in which about two thirds of the grid blocks (1761) are active. The grid blocks have equal 180 meter sides in x and y

directions. The reservoir simulation case has been modeled with corner point geometry and a Carter-Tracy aquifer. The complete data set for this reservoir is available online [PUNQ 2010]. The geological parameters varied are the horizontal and vertical permeabilities and the porosities.

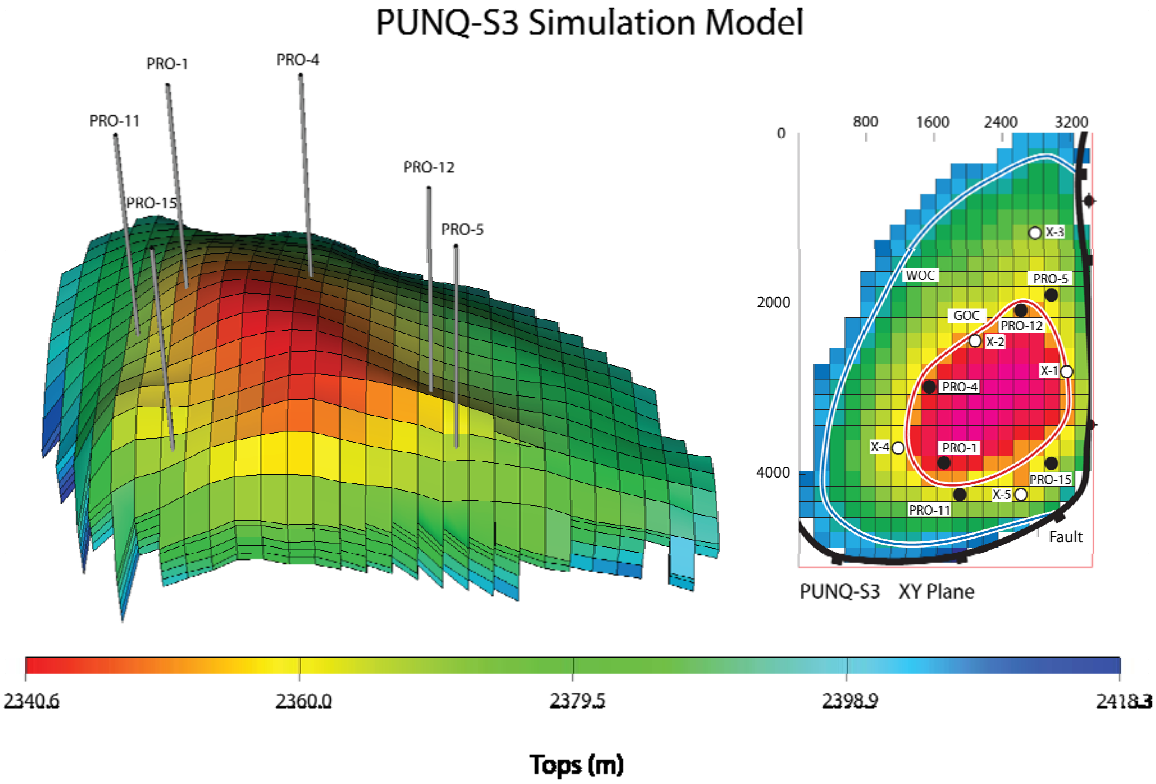

Figure 1: PUNQ-S3 reservoir model with top surface map and well positions

The production history of the six wells is summarized as the following periods:

1- An extended well testing period during the first year (four three-monthly production periods).
2- Shut-in period for the following three years
3- 12 year production period with fixed production rate at 150 sm^3/day
4- Shut in for each well for a period of two weeks in each production period for the purpose of testing the well.

The simulated production history for the first 8 years from six wells was generated by the Netherlands Organization for Scientific Research (TNO). The production history included pressure, water cuts and gas-oil ratios for each well. In order to reflect the real world measurement errors, Gaussian noise was added to the well porosity/permeability and to the synthetic field production data. The standard deviation of noise on poro/perm values was set to 15 %. The noise for production data was correlated in time to mimic the systematic character of errors in such data. The noise level on the shut-in pressures was 3 times smaller than on the flowing pressure, to reflect the more accurate shut-in pressure measurements. The noise added to the gas oil ratio (GOR) data was set at 10% before gas breakthrough and 25% after gas breakthrough, indicating the difference between the solution and the free gas situation. Also for well watercut (WCT) data, we have noise at 2% before and 5% after water breakthrough.

After 8 years of production, there were two different recovery scenarios. The first strategy was to continue the current production for another 8.5 years with the same 6 or wells and the second alternative was to add 5 new wells (X1-X5) which their position has been marked with white dots on figure 1 and continue the production for the total period of 16.6 years. In this chapter we only consider the first scenario: continuing the production with the original wells. The wells operate under a production constraint. After falling below a limiting bottom hole pressure, wells switch to the BHP constraint.

We used different algorithms to minimize the following objective function in PUNQ-S3 case [Barker et al. 2001]:

$$SoS(o^{obs}, p) = \frac{1}{n_w}\sum_i \frac{1}{n_p}\sum_j \frac{1}{n_t}\sum_k \left(w_{ijk}\, \frac{o_{ij}^{obs}(t^k) - o_{ij}^{sim}(t^k; p)}{\sigma_{ijk}} \right)^2 \quad \text{.....} \qquad (5)$$

where n_w is the number of wells with subscript i running over wells, n_p is the number of production data types with subscript j running over them. Subscript k runs over production data report times and n_t is the respective number of samples. Observed data

(o^{obs}) (bottomhole pressures, gas oil ratio and watercuts) and simulated ones (o^{sim}) for each of the parameters (p) were reported at time steps t^k with a measurement error of σ. At each time step for the parameters, there were extra weighting factors denoted with w. These weights reflect the importance of some of data types at specific time steps and are indicated in the online dataset provided [PUNQ-S3 model 2010]. Tables 1-3 summarize the weights and sigma values used in the calculation of history matching misfit in the PUNQ-S3 reservoir. As can be seen in these tables, measurement errors (σ) and weights (w) are different for each variable type and time step. For example, there are larger sigma values for the gas oil ratio. Also, there were different number of measurements for variables used in the objective function; for example we had more frequent measurements of bottom hole pressures.

Table 1: Summary of the weights and sigma values for BHP data used in the misfit calculation from [PUNQ 2010]

BHP	PRO-1: PRO-15	
Time	Sigma	Weight
1	3	1
91	3	1
182	3	1
274	3	1
366	1	1
1461	1	1
1826	3	1
1840	1	1
2192	3	1
2206	1	1
2557	3	1
2571	1	1
2922	3	1
2936	1	1

Table 2: Summary of the weights and sigma values for GOR data used in misfit calculation from [PUNQ 2010]

GOR	PRO-1		PRO-4		PRO-5		PRO-11		PRO-12		PRO-15	
Time	Sig	Wgt	Sig	Wgt	Sig	Wgt	Sig	Wgt	Sig	Wgt	Sig	Wgt
1642	7.3	1	-	-	-	-	-	-	-	-	-	-
1826	33.7	4	-	-	6.4	1	6.7	1	6.7	1	6.4	1
1841	-	-	8.2	1	-	-	-	-	-	-	-	-
2008	47.9	1	41.3	4	-	-	-	-	-	-	-	-
2192	-	-	-	-	6.2	1	5.9	1	7.4	1	5.6	1
2378	36.8	1	26.5	1	-	-	-	-	-	-	-	-
2557	-	-	-	-	6.2	1	6.2	1	6.8	1	5.8	1
2738	47.5	1	7.5	1	-	-	-	-	-	-	-	-
2922	-	-	-	-	6.5	1	6.5	1	7.6	1	5	1

Table 3: Summary of the weights and sigma values for WCT data used in misfit calculation from [PUNQ 2010]

WCT	PRO-1		PRO-4		PRO-5		PRO-11		PRO-12		PRO-15	
Time	Sig	Wgt	Sig	Wgt	Sig	Wgt	Sig	Wgt	Sig	Wgt	Sig	Wgt
2572	-	-	-	-	-	-	0.02	4	-	-	-	-
2738	-	-	-	-	-	-	0.02	4	-	-	-	-
2922	0.02	4	0.02	4	0.02	4	0.05	4	0.02	4	0.02	4

5.1.3 PUNQ-S3 Parameterization

For the parameterization of the reservoir model, 3 different setups with homogeneous regions were designed in this work: Case A, B and C. These setups differ in the number of regions and unknown parameters defined in the history matching process.

The initial ranges for unknown parameters are given in Table 4. These values are based on the geological description of the reservoir. Layers 1, 3 and 5 are high quality and layers 2 and 4 have lower porosity and permeability values.

Table 4: Initial ranges for parameters in PUNQ-S3 reservoir

Layer	Porosity	Horizontal Permeability (mD)	Vertical permeability (mD)
1	0.15-0.3	133-3013	44-925
2	0.05-0.15	16-133	8-44
3	0.15-0.3	133-3013	44-925
4	0.1-0.2	47-376	17-118
5	0.15-0.3	133-3013	44-925

In setup A, there are 5 layers in the model. Each layer has homogenous properties (porosity, horizontal and vertical permeability), resulting in 15 unknown parameters that should be estimated. Ant colony optimization (ACO_R) was used to estimate these unknown values using different tuning parameters. A summary of the ACO_R parameters used in these trials is given in Table 5. The best misfit obtained in this setup was 6.29 for the A-1 case. As can be seen in the minimum misfit column of Table 5, this setup does not provide a good match to the results from PUNQ-S3 model.

Table 5: Parameters of ACO_R and misfit values obtained for setup A

Case	Number of ants	k	q	Ξ	Total simulations	Misfit
A-1	100	300	0.1	0.5	2000	6.29
A-2	25	200	0.1	0.5	1500	6.59
A-3	50	100	0.01	0.5	1000	6.76
A-4	50	200	0.2	0.4	1500	6.84

In setup B, we had 5 layers and each layer was divided into 3 regions. Unknown porosity, vertical and horizontal permeability values were considered the same in each

of the 3 regions within each layer. This raised the total number of unknowns to 45 parameters. Table 6 shows the parameter values and the misfits for the case B setup with a minimum misfit value of 3.47 obtained in case B-1.

Table 6: Parameters of ACO_R and misfit values obtained for setup B

Case	Number of ants	k	q	ξ	Total simulations	Misfit
B-1	50	500	0.05	0.2	3000	3.47
B-2	100	1000	0.01	0.3	5000	3.60
B-3	100	500	0.01	0.1	3000	5.92
B-4	50	500	0.001	0.1	2000	4.82
B-5	100	500	0.01	0.15	3000	4.58
B-6	50	500	0.005	0.15	2000	3.77
B-7	50	500	0.001	0.3	3000	3.82

In setup C, we parameterized the PUNQ-S3 model with five layers and nine homogenous regions per layer. We used ant colony optimization, differential evolution and neighbourhood algorithms to estimate the porosities in each homogenous region and layer of the reservoir. Five layers times nine regions per layer makes 45 porosity values that should be estimated in the assisted history matching framework. From least square fitting of the well data crossplots, the following deterministic relationships were obtained in a study performed by Total (Elf Exploration) [Boss, 1999]. These correlations assumed a linear relationship between porosity and logarithmic horizontal permeability and between horizontal and vertical permeability.

$$\ln(k_h) = 0.77 + 9.03\phi \quad \text{.....} \quad (6)$$

$$k_v = 3.124 + 0.306k_h \quad \text{.....} \quad (7)$$

After estimating 45 porosity values, we used the above correlations to determine the remaining horizontal and vertical permeability values in the regions and layers of the

reservoir. These relationships were used in order to reduce the number of unknown parameters.

To test the initial setup, like case B, we used the ant colony optimization algorithm to find history-matched models. The parameters used for the ACO_R algorithm and the best misfit values are summarized in Table 7. For each case, the tuning parameters and maximum number of simulations are same in both B and C setups. The best misfit in this setup was 2.28 for case C-2.

Table 7: Parameters of ACO_R and misfit values obtained for setup C

Case	Number of ants	k	q	ξ	Total simulations	Misfit
C-1	50	500	0.05	0.2	3000	2.75
C-2	100	1000	0.01	0.3	5000	2.28
C-3	100	500	0.01	0.1	3000	4.39
C-4	50	500	0.001	0.1	2000	6.57
C-5	100	500	0.01	0.15	3000	3.99
C-6	50	500	0.005	0.15	2000	4.78
C-7	50	500	0.001	0.3	3000	3.47

In order to investigate how the choice of parameterization can affect the final misfit value and the match quality of the models, the ACO_R run parameters were set to be identical for runs in cases B and C. We ignored the effect of random seed in this comparison. Figure 2 shows the minimum misfits obtained in seven cases of B and C setups. As can be seen from this figure, in 5 trials, the case C setup (which has more regions per layer) resulted in a better final misfit value.

Based on the results of three different homogenous layer parameterizations, it was decided to take the setup C as the choice of base model to compare different agent-based optimization algorithms. In section 2, we compare ant colony optimization, differential evolution and the neighbourhood algorithms for history matching of the PUNQ-S3 model with the above parameterization.

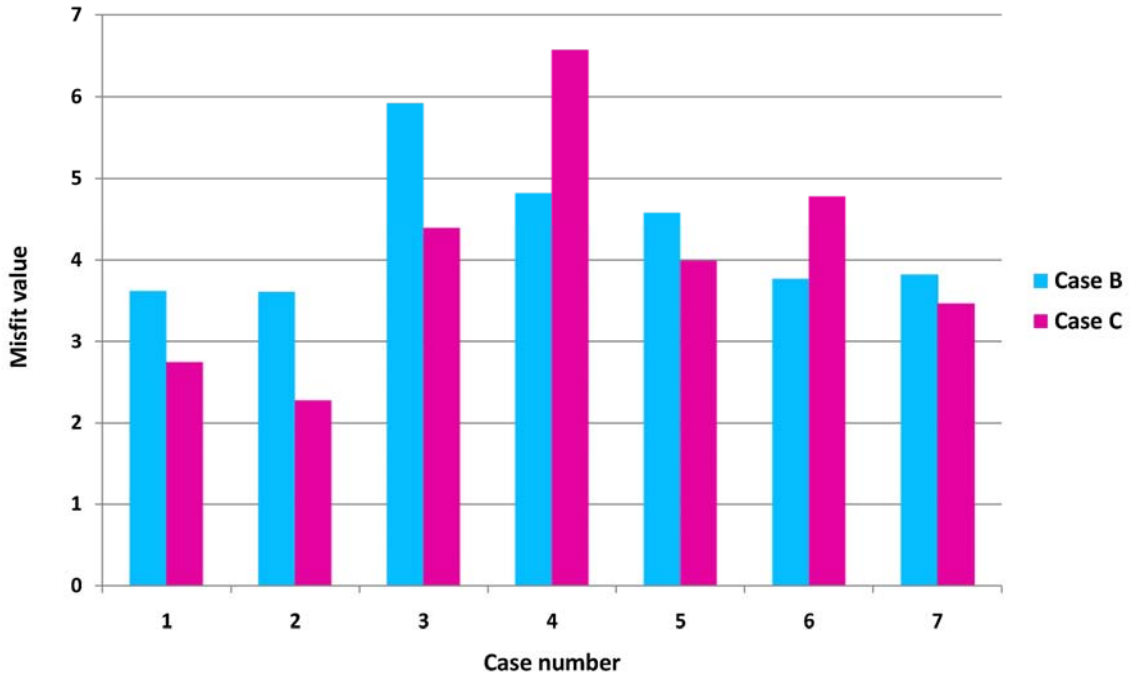

Figure 2: Comparison between minimum misfits in setups B and C

5.2 History Matching Results

In this part, a comparative study of ant colony optimization, differential evolution and the neighbourhood algorithm is presented. In all runs, the stopping criteria for the algorithms were defined as the maximum number of simulations which was set to be 3000. Running 3000 simulations of the PUNQ-S3 model, depending on the selected node and input parameters provided by the optimization algorithm, took between 22 and 30 hours. Since the neighbourhood algorithm runs in parallel, the required run time for these simulations was less than for the ant colony optimization and differential evaluation algorithms. Simulations were performed based on the guidelines developed in chapter 4 for tuning parameters. We use ECLIPSE software to perform flow simulations.

5.2.1 Ant Colony Optimization (ACO_R) Runs

For the ant colony optimization runs, we used different archive sizes (k), number of ants (m), search localization (q) and pheromone evaporation rates (ξ). Cases ACO-1 to ACO-20 have the same number of ants and archive size. As demonstrated in chapter 4, this

setting provided good results for the Teal South example. By setting *m=k*, the whole archive will be updated at the end of each iteration using the same number of ants. Table 8 presents the tuning parameters of ACO_R algorithm and the best misfit values in each case. In these cases, search localization (q) and pheromone evaporation rates (ξ) were selected to be identical for archive size and number of ants of 50 and 100 (i.e. ACO-1 and ACO-11 have same q and ξ values).

Table 8: Parameters of ACO_R and misfit values obtained for different setups (*m=k*)

Case	*m*	*k*	*q*	ξ	Total simulations	Misfit
ACO-1	50	50	0.5	0.5	3000	2.06
ACO-2	50	50	0.9	0.9	3000	2.41
ACO-3	50	50	0.5	0.9	3000	2.26
ACO-4	50	50	0.9	0.5	3000	2.24
ACO-5	50	50	0.7	0.7	3000	1.90
ACO-6	50	50	0.7	0.5	3000	2.56
ACO-7	50	50	0.8	0.3	3000	3.80
ACO-8	50	50	0.8	0.5	3000	2.29
ACO-9	50	50	0.4	0.7	3000	1.83
ACO-10	50	50	0.6	0.8	3000	2.46
ACO-11	100	100	0.5	0.5	3000	3.42
ACO-12	100	100	0.9	0.9	3000	2.31
ACO-13	100	100	0.5	0.9	3000	3.05
ACO-14	100	100	0.9	0.5	3000	2.71
ACO-15	100	100	0.7	0.7	3000	2.89
ACO-16	100	100	0.7	0.5	3000	3.04
ACO-17	100	100	0.8	0.3	3000	3.42
ACO-18	100	100	0.8	0.5	3000	3.09
ACO-19	100	100	0.4	0.7	3000	2.83
ACO-20	100	100	0.6	0.8	3000	2.97

Analyzing the best misfit values obtained in Table 8, an average misfit value of 2.38 was obtained for ACO-1 to 10 cases, while this value for cases ACO-11 to 20 was 2.97. When history matching of PUNQ-S3 using the ACO_R algorithm, and setting the number

of ants equal to the archive size, we concluded that smaller values for these two parameters are more helpful in finding lower final misfits. Although a larger number of ants may contribute to a wider exploration of the search space, in this high dimensional problem, this setting may result in reaching the maximum number of simulations without full exploration of the search space. Having more iterations by using fewer ants per generation helps towards maturity of the search, allowing the algorithm to explore and converge to good-fitting regions of the search space with limited number of simulations.

To test the effect of parameterization on the misfit value, seven initial tests were performed in section 1 of this chapter using the ACO_R algorithm. Further to these tests which are reported in Table 7, three more cases were considered to see if a larger archive size helped to obtain lower misfit values. Table 9 reports the tuning parameters and best misfit values obtained in these additional tests.

Table 9: Parameters of ACO_R and misfit values obtained for different setups ($m \neq k$)

Case	m	k	q	ξ	Total simulations	Misfit
ACO-21	50	100	0.5	0.5	3000	2.72
ACO-22	50	200	0.5	0.9	3000	3.1
ACO-23	50	300	0.5	0.5	3000	2.47

Considering the results reported in Table 7 and additional tests in Table 9, where larger archive sizes were used in the ACO_R algorithm, we can see that for this problem, a large archive size does not necessarily result in a better final misfit value. Cases ACO-21 and ACO-1 have the same number of ants and tuning parameters, while ACO-21 has a larger archive size. In fact, among several trials, none of the runs with a large archive size could obtain a misfit value less than 2 although we had two misfit values under 2 (ACO-5 = 1.9 and ACO-9 = 1.83) for a small number of ants and archive size. This might be due to the fact that the ACO_R algorithm works by ranking of models, placing good models at the top of the archive. This mechanism, in combination with low to middle search localization values (q), puts a strong weight on the best fitting models at

the top of the archive and may withdraw attention of the algorithm from other members of archive, even when a large archive is used. However, on the other hand, using a large archive size may help in better exploration of the search space at initial stages of sampling. Based on the above results, we notice that ACO-9 case has the best misfit (M=1.83) among different trials. We will use this case for further comparisons with other techniques.

5.2.2 Differential Evolution (DE) Runs

In history matching the PUNQ-S3 model, we considered four strategies of the differential evolution algorithm. These strategies included DE-Rand, DE-Best, DE-Rand-Best and DE-Best-2. Like the runs with the ACO_R algorithm, in the differential evolution algorithm, the maximum number of simulations (i.e. 3000) was considered as the stopping criterion.

Tables 10-13 present the tuning parameters and best misfit values obtained for different cases in each strategy of differential evolution. We have tried different population sizes (N_p), scaling factor (F) and crossover values (C_r) in each strategy. These values are identical for the same case number in each strategy (i.e. DE-1 has same tuning parameters in all strategies).

Table 10: DE-Rand runs with tuning parameters and best misfit in each case

Case	N_p	F	C_r	Total simulations	Misfit
DE-1	100	0.3	0.5	3000	3.06
DE-2	100	0.5	0.5	3000	3.33
DE-3	50	0.3	0.5	3000	2.35
DE-4	50	0.5	0.5	3000	2.11
DE-5	100	0.7	0.5	3000	4.08
DE-6	50	0.7	0.5	3000	2.19
DE-7	100	0.5	0.7	3000	3.35
DE-8	150	0.5	0.5	3000	3.41
DE-9	50	0.5	0.7	3000	1.95
DE-10	100	0.7	0.3	3000	2.75

Table 11: DE-Best runs with tuning parameters and best misfit in each case

Case	N_p	F	C_r	Total simulations	Misfit
DE-11	100	0.3	0.5	3000	1.80
DE-12	100	0.5	0.5	3000	1.49
DE-13	50	0.3	0.5	3000	2.97
DE-14	50	0.5	0.5	3000	1.45
DE-15	100	0.7	0.5	3000	1.64
DE-16	50	0.7	0.5	3000	1.59
DE-17	100	0.5	0.7	3000	1.48
DE-18	150	0.5	0.5	3000	1.51
DE-19	50	0.5	0.7	3000	1.49
DE-20	100	0.7	0.3	3000	1.90

Table 12: DE-Rand-to-Best runs with tuning parameters and best misfit in each case

Case	N_p	F	C_r	Total simulations	Misfit
DE-21	100	0.3	0.5	3000	3.39
DE-22	100	0.5	0.5	3000	2.24
DE-23	50	0.3	0.5	3000	3.38
DE-24	50	0.5	0.5	3000	1.69
DE -25	100	0.7	0.5	3000	2.33
DE -26	50	0.7	0.5	3000	1.42
DE -27	100	0.5	0.7	3000	2.51
DE -28	150	0.5	0.5	3000	1.80
DE -29	50	0.5	0.7	3000	1.87
DE -30	100	0.7	0.3	3000	1.69

Table 13: DE-Best-2 with tuning parameters and best misfit in each case

Case	N_p	F	C_r	Total simulations	Misfit
DE-31	100	0.3	0.5	3000	1.88
DE-32	100	0.5	0.5	3000	2.49
DE-33	50	0.3	0.5	3000	1.41
DE-34	50	0.5	0.5	3000	1.69
DE-35	100	0.7	0.5	3000	3.1
DE-36	50	0.7	0.5	3000	2.79
DE-37	100	0.5	0.7	3000	2.09
DE-38	150	0.5	0.5	3000	1.85
DE-39	50	0.5	0.7	3000	1.63
DE-40	100	0.7	0.3	3000	2.56

From the above tables, we see that in the DE-Rand strategy, the best misfit value is 1.95 for DE-9 run, in DE-Best we have a best misfit of 1.45 for the DE-14 case, for DE-Rand-Best, the setup for DE-26 gives the best misfit of 1.42 and the best misfit obtained in DE-Best-2 strategy is 1.41 in DE-33 case. These cases are selected for further comparisons in next section.

5.2.3 Neighbourhood Algorithm (NA) Runs

For the neighbourhood algorithm runs, we needed to consider two different situations regarding the number of initial samples and sampled cells in each iteration. Ant colony optimization (ACO_R) setups had both trials with number of ants equal to archive size and trials with larger archive size in comparison with the number of ants. For NA runs to be in harmony with ACO_R trials where the archive size is larger than the number of ants, we launched the first set of history matching runs using similar configurations for the NA algorithm. In these experiments, we have two different initial population sizes (n_{si}=500 and n_{si}=1000) and we have tried two different values for the number of cells to be sampled (n_s=50 and n_s=100). On the other hand in the NA algorithm, as previously

mentioned, the ratio of sampled to resampled cells (n_s/n_r) in each iteration controls behavior of the algorithm. If $n_s = n_r$, NA has its most explorative setting, where it performs a more global search, instead of over-refining previously found regions. Based on this fact, we also used different n_r values for each category of runs in the neighbourhood algorithm. Having the opportunity to launch parallel jobs of the neighbourhood algorithm, for each setup of the algorithm, we tried three different runs, each having different initial seed values. Results of best misfit values for the first set of history matching runs using the neighbourhood algorithm are shown in figure 3.

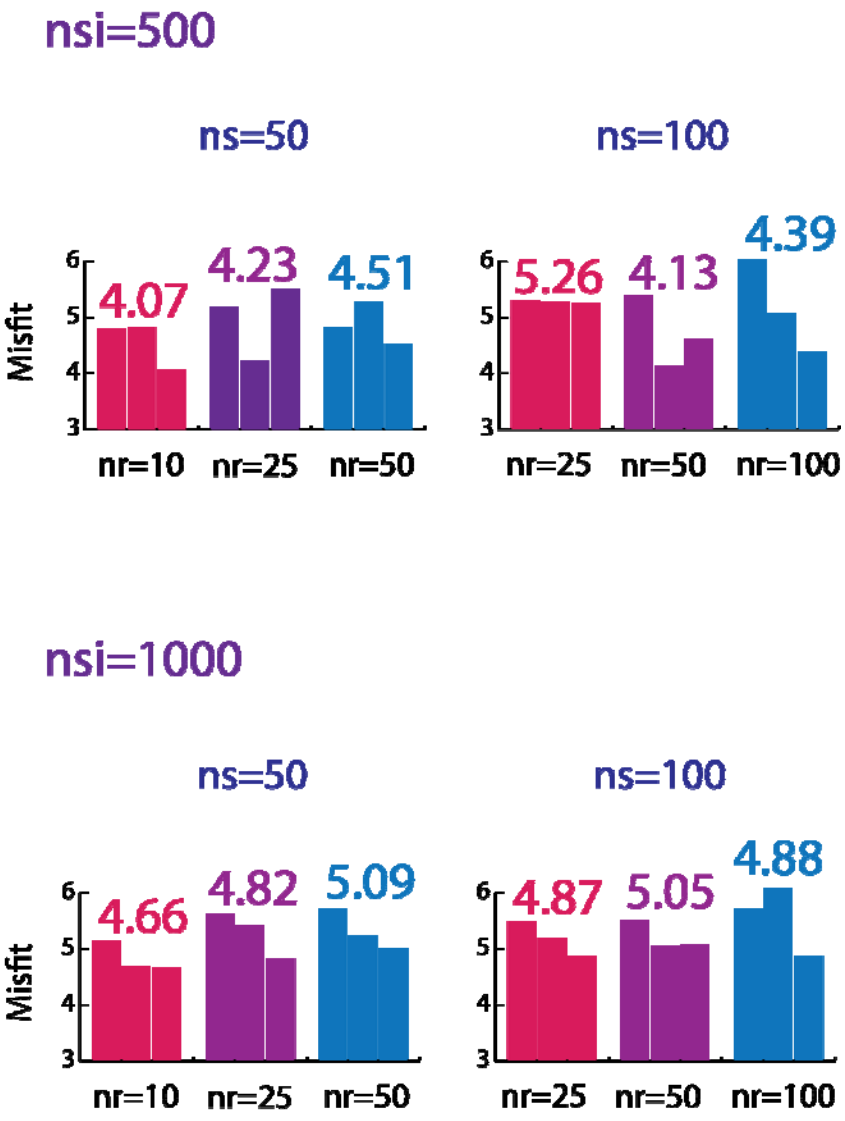

Figure 3: Best misfit results for different tuning parameters of NA similar to ACO_R setups with archive size larger than number of ants

The second series of history matching runs using the neighbourhood algorithm were tied to the setup of the ACO_R algorithm where we have the same number of ants and archive size ($m = k$) and also the differential evolution experiments where a constant population size was used in all iterations. To have a similar algorithm setup in NA, we set the initial sample size to be equal to the number of cell beings sampled in each iteration (i.e. $n_{si} = n_s$). Figure 4 presents the best misfit results in each setup of neighbourhood algorithm using different n_r values.

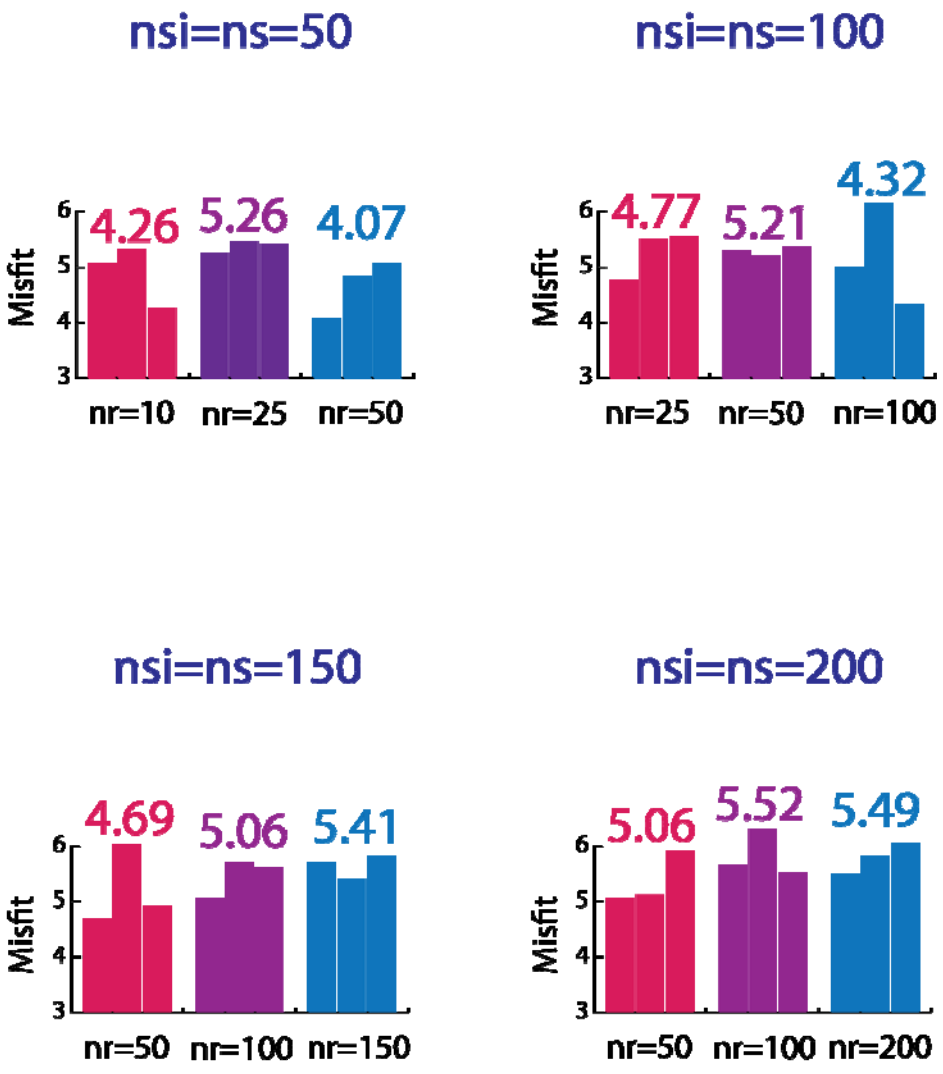

Figure 4: Best misfit results for different tuning parameters of NA similar to DE setups and ACO_R trials with archive size equal to number of ants

Looking at the results of the neighbourhood algorithm applied to history matching of the PUNQ-S3 model, we can see that the best misfit value in the above tests was 4.07 for the case that n_{si}=n_s=50 and n_r=50. It should be noted that we also had a misfit value of 4.07 for the setup when we use an initial population of 500 and number of sampled cells equal to 50 (Figure 3). However since these simulations have a large initial population (hence taking longer to converge) and best misfit models in ACO_R and DE setups had a population size of 50 per iteration, we take NA runs with equal population size in each iteration (n_{si}=n_s=50) for further comparison with other algorithms. Table 14 summarizes the tuning parameters for the two best NA runs obtained from above experiments with misfit values of 4.07 and 4.26 respectively.

Table 14: Tuning parameters and best misfit obtained for neighbourhood algorithm (NA)

Case	n_{si}	n_s	n_r	Generations	Best Misfit
NA-1	50	50	50	60	4.07
NA-2	50	50	10	60	4.26

5.2.4 Neighbourhood Algorithm Coupled with Geostatistical Framework

Using homogenous layers is not the only way of parameterizing a reservoir simulation model in history matching. Demyanov [2003] used a geostatistical framework to perturb the reservoir properties in history matching of PUNQ-S3. He used two setups in his work: a full model with 60 free parameters and a reduced model with 20 unknown parameters. The neighbourhood algorithm was used to sample the geostatistical model parameter space (variogram properties such as sill/nugget, angle, etc) in these two setups. In this framework, the neighbourhood algorithm proposes a set of parameters for variogram properties and resulted reservoir models based on geostatistical parameterization are flow simulated to obtain misfit values. Table 15 summarizes the results obtained by Demyanov using different numbers of unknown parameters and tuning parameters for the neighbourhood algorithm.

Table 15: Summary of NA runs in the geostatistical framework with full and reduced models [Demyanov, 2003]

NA parameters				Number of Models	Best model index	Best misfit	Number of	
Init size	Sample size	Iteration number	Resamp number				layers	parameters
100	80	100	60	8100	7621	8.37	5	60
200	100	200	50	20200	9040	7.94	5	60
200	100	150	10	15200	13536	8.65	5	60
100	70	100	20	7100	6309	7.20	5	20
50	30	50	15	1550	1402	8.67	5	20
200	100	200	50	20200	19243	3.05	5	20
500	200	200	75	40500	10950	2.99	5	20

As it can be seen from above table, the best misfits using 20200 simulations were 7.93 for the full model with 60 unknown parameters and 3.05 for the reduced model with 20 unknown parameters. A lower misfit with the reduced model does not necessarily indicate the advantage of this setup, but may be an indication of poor sampling of the search space in the full model with 60 unknown parameters. We take the best misfit values in two setups using 20200 simulations (7.94 in full setup and 3.05 in reduced model) for comparison with ant colony optimization, differential evolution and direct application of the neighbourhood algorithm for history matching of the PUNQ-S3 model.

5.2.5 Comparison of Best Misfit and Their Fit to Production Data

After running multiple tests with ant colony optimization (ACO_R), four strategies of differential evolution (DE) and the neighbourhood algorithm (NA), we can now compare their performance in terms of the best misfit value and the quality of history matched models in capturing the historical production data available from the PUNQ-S3 reservoir. Figure 5 compares the best misfit values for various algorithms used in this work and also the setup proposed by Demyanov [2003].

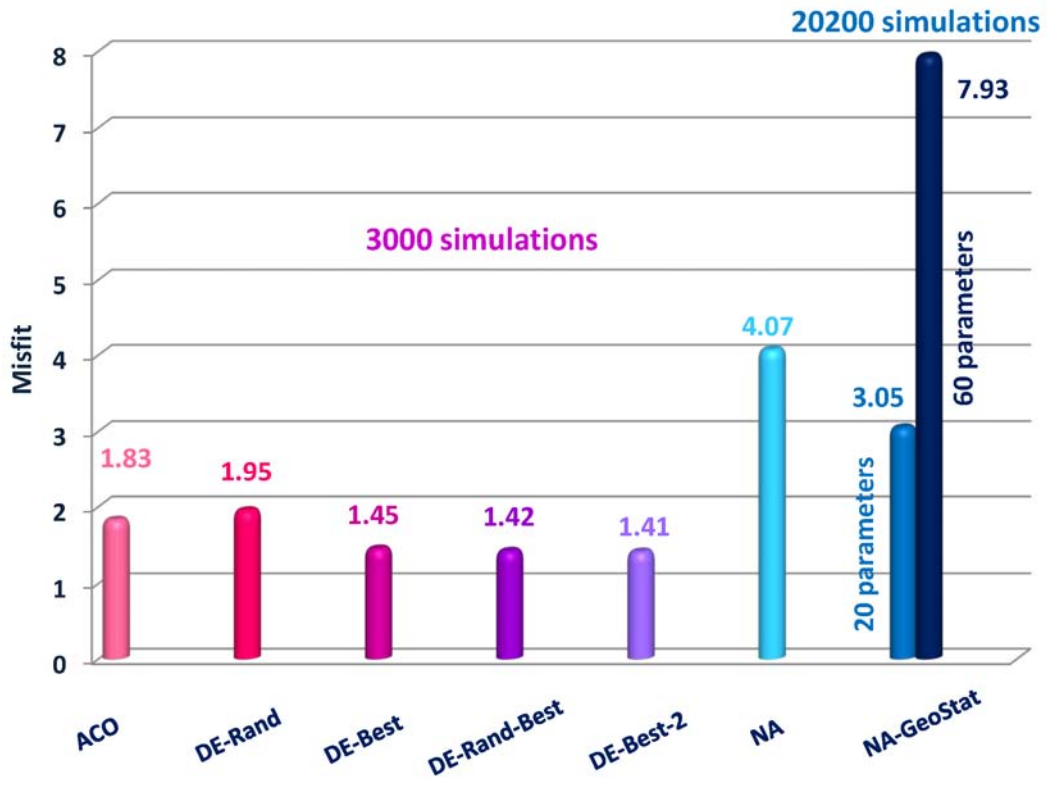

Figure 5: Comparison of best misfits obtained for different algorithms

From the best misfits reported in the above figure, we see that DE and ACO_R algorithms, in general, provided good history matching results. The best result was obtained using DE-Best-2 strategy, followed by DE-Rand-Best and DE-Best strategies with marginal difference. The performance of the NA was not satisfactory in this high dimensional history matching problem, providing poor match results. Extreme exploration NA settings (NA-1) which performed the widest search obtained a better misfit in comparison with a less explorative NA setting (NA-2). The poor result of NA-2 is possibly due to over-refinement of local minima. On the other, we can see that using 20200 simulations, the best misfit value obtained with the geostatistical framework is 3.05. Comparing this value with the results of ACO_R and DE algorithm applied for the PUNQ-S3 problem with same misfit definition and 3000 simulations, we can see a good improvement in both final misfit value and number of required simulations.

Figures 6-12 show the best history matching results for selected wells in PUNQ-S3 model during history (days 0-2936) and prediction periods (days 2936-6025). These

figures show the quality of match for bottom hole pressure (BHP) in wells 1,4 and 5, gas oil ratio (GOR) for wells 1, 4 and 11 and water cut (WWC) in well 11. In these figures yellow squares with red borders show the observed data used to calculate misfit values during history matching.

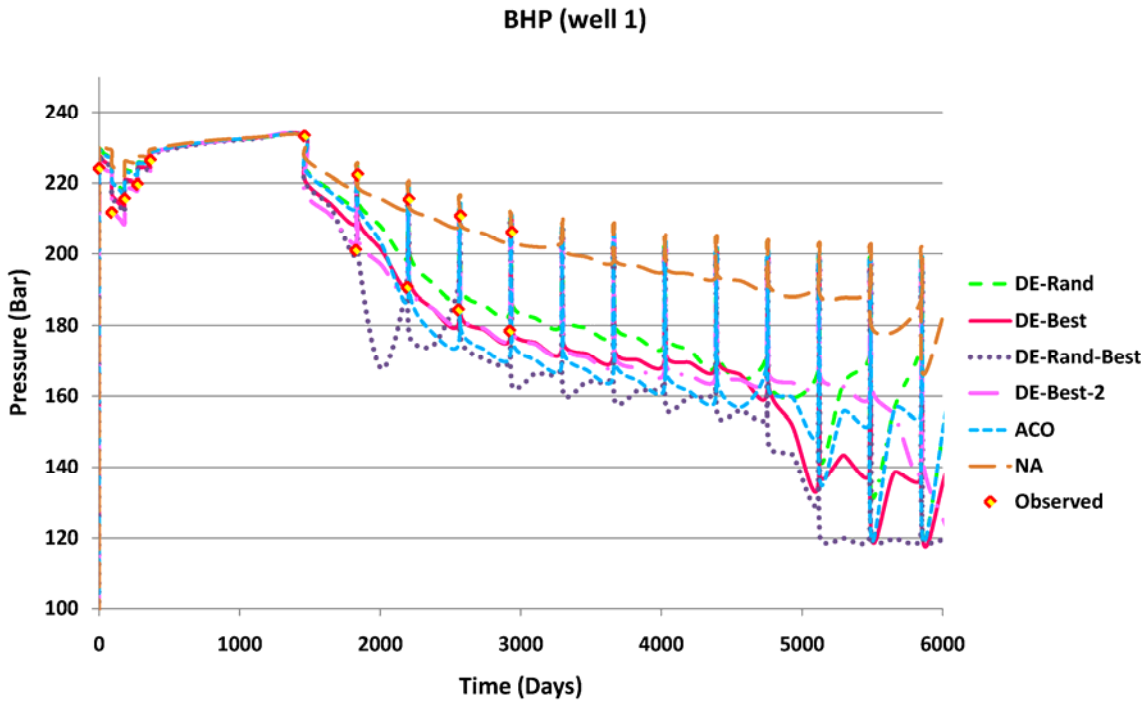

Figure 6: BHP match result for well 1

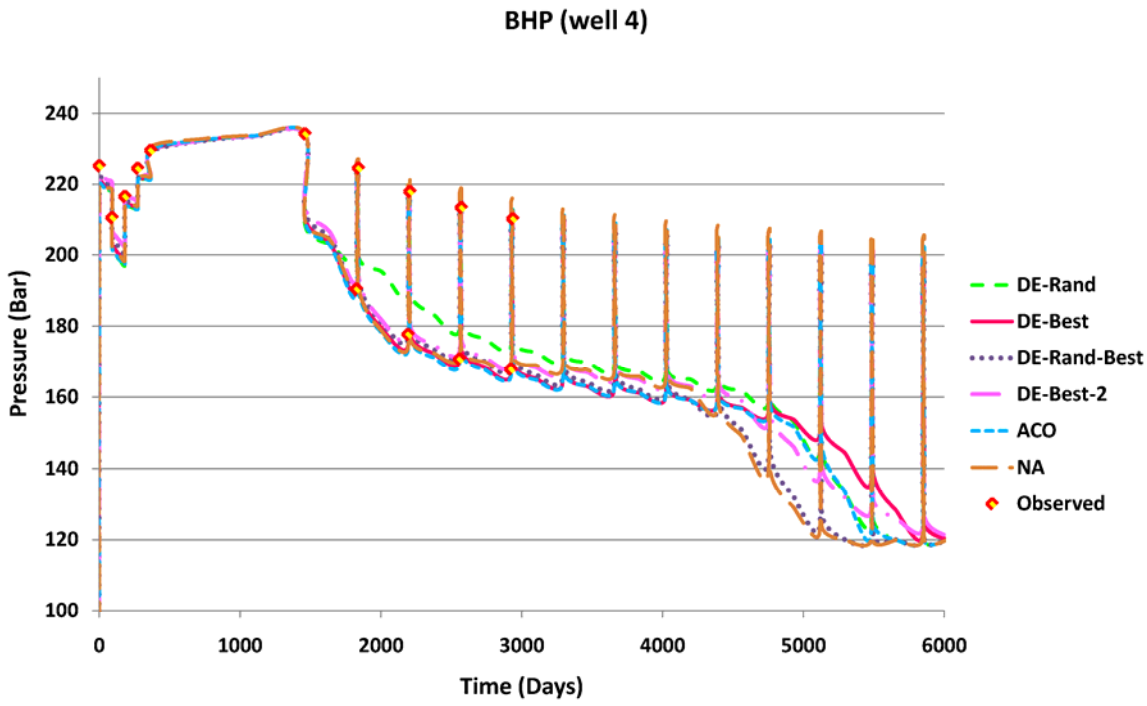

Figure 7: BHP match result for well 4

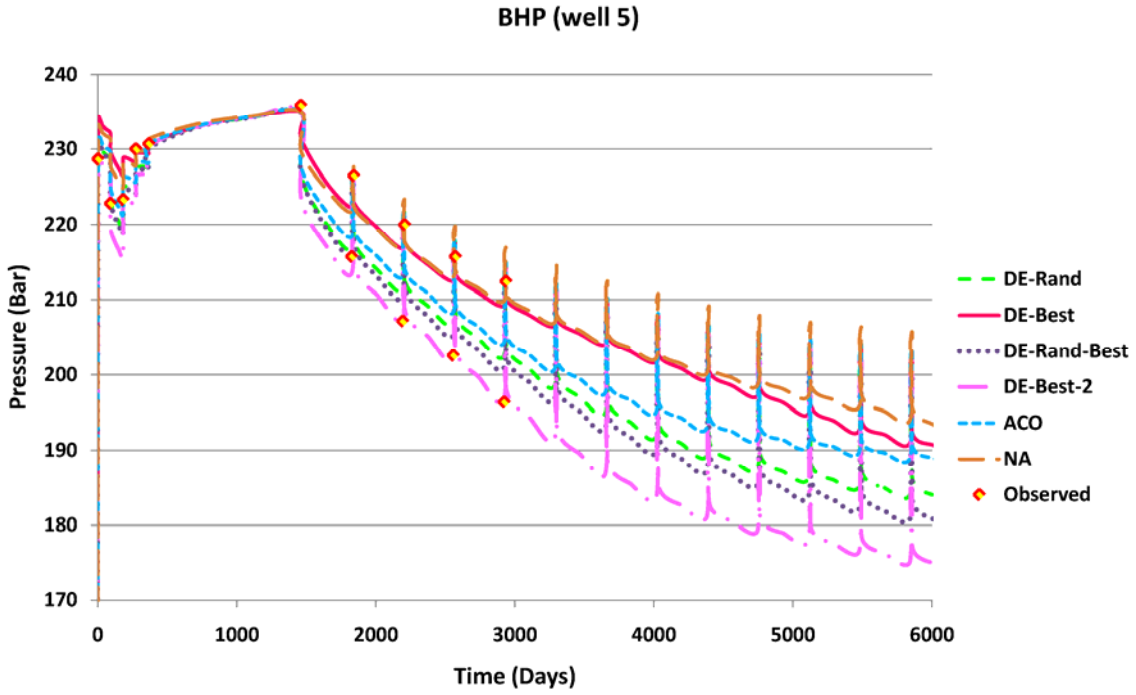

Figure 8: BHP match result for well 5

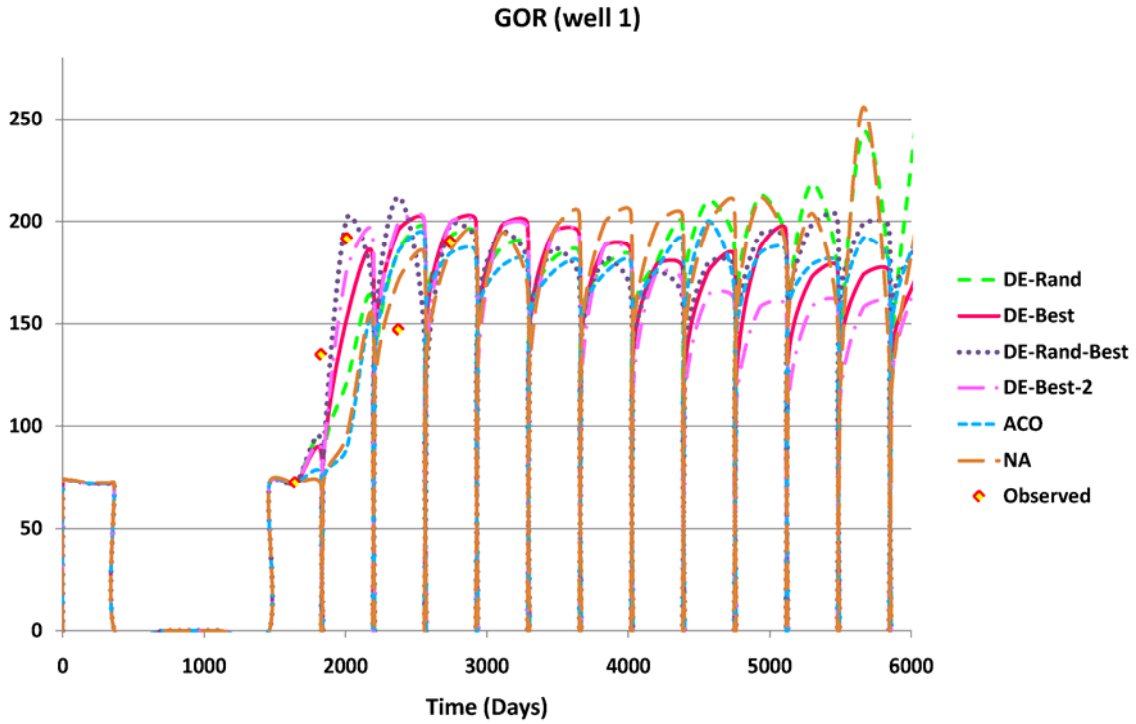

Figure 9: GOR match result for well 1

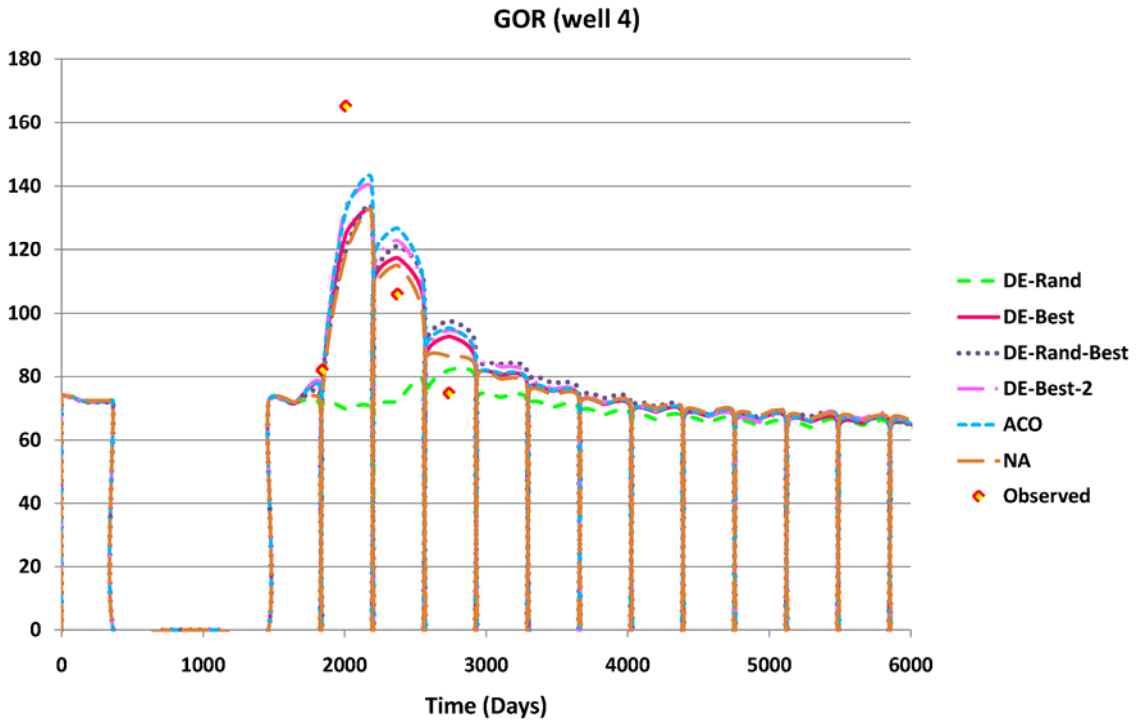

Figure 10: GOR match result for well 4

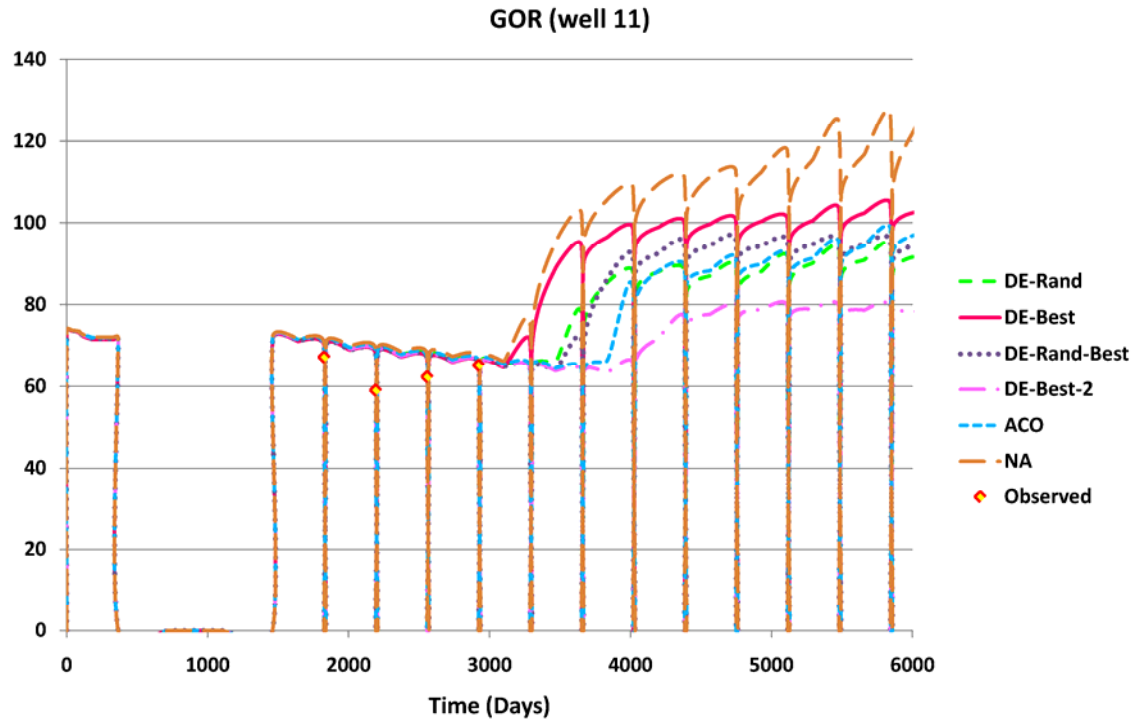

Figure 11: GOR match result for well 11

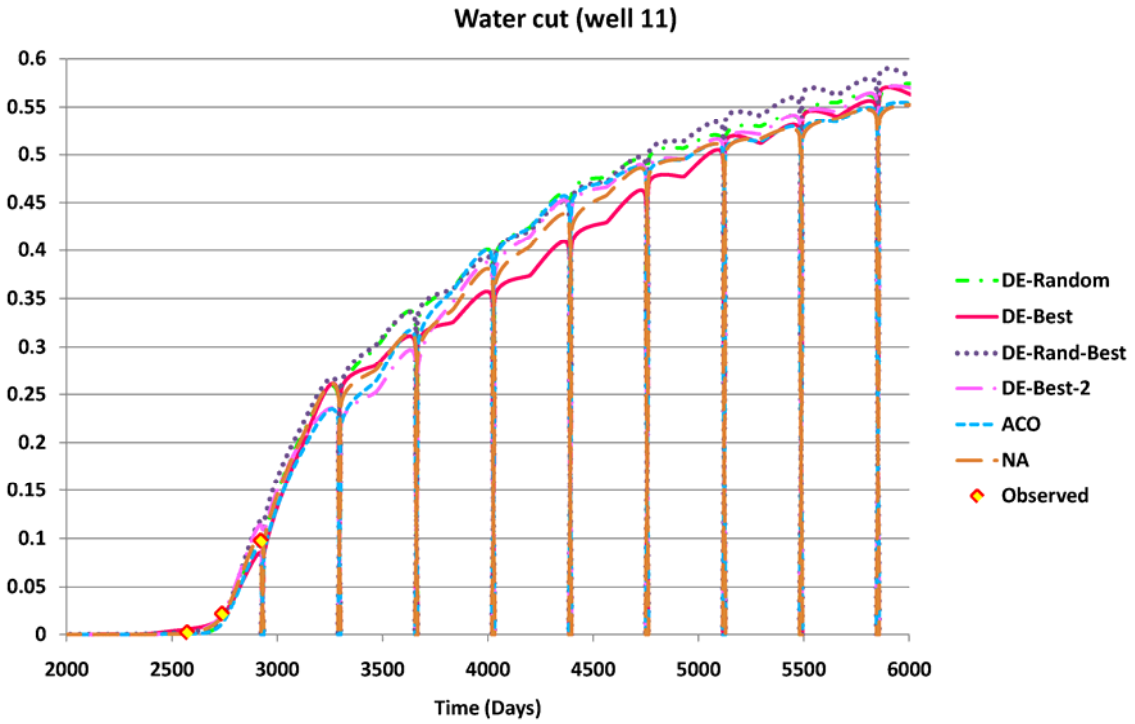

Figure 12: WWC match result for well 11

Examining figures 6-12, we see that ACO_R and DE algorithms provided a reasonable match over the history period. One interesting observation in these figures is the deviation in behavior of various variables, such as BHP in well 1 and 5 and GOR in well 5 in the forecast period. In section 2.7 by looking to sampling performance of algorithms, we will analyze the different behavior of the algorithms in the history matching and forecast periods. In the above figures, as expected, NA match quality was not satisfactory in various elements of available production data, for example BHP in well 1.

Figures 6-12 show that a single best history match may not be a good predictor [Tavassoli et al. 2004]. While DE-Best, DE-Rand-Best and DE-Best-2 strategies have almost the same misfits, they demonstrate different behavior in the forecast period. Still many companies in our industry take one or two "best" match results and use them to predict the future performance of the reservoir. This research demonstrates the benefit of using multiple history-matched models in reservoir predictions. In section 3, we will demonstrate the benefits of such a approach in making reliable forecasts.

5.2.6 Comparison of Convergence Speeds

In this section, we compare the efficiency of different algorithms in reducing misfit values in the history matching of the PUNQ-S3 reservoir. For this purpose, we have selected the setups of algorithms resulting in the minimum misfits. Figure 13 plots the best misfits obtained in each generation of the algorithms versus the generation number. In all algorithms, each generation has 50 members which, when considering 3000 simulation, makes 60 generations in total.

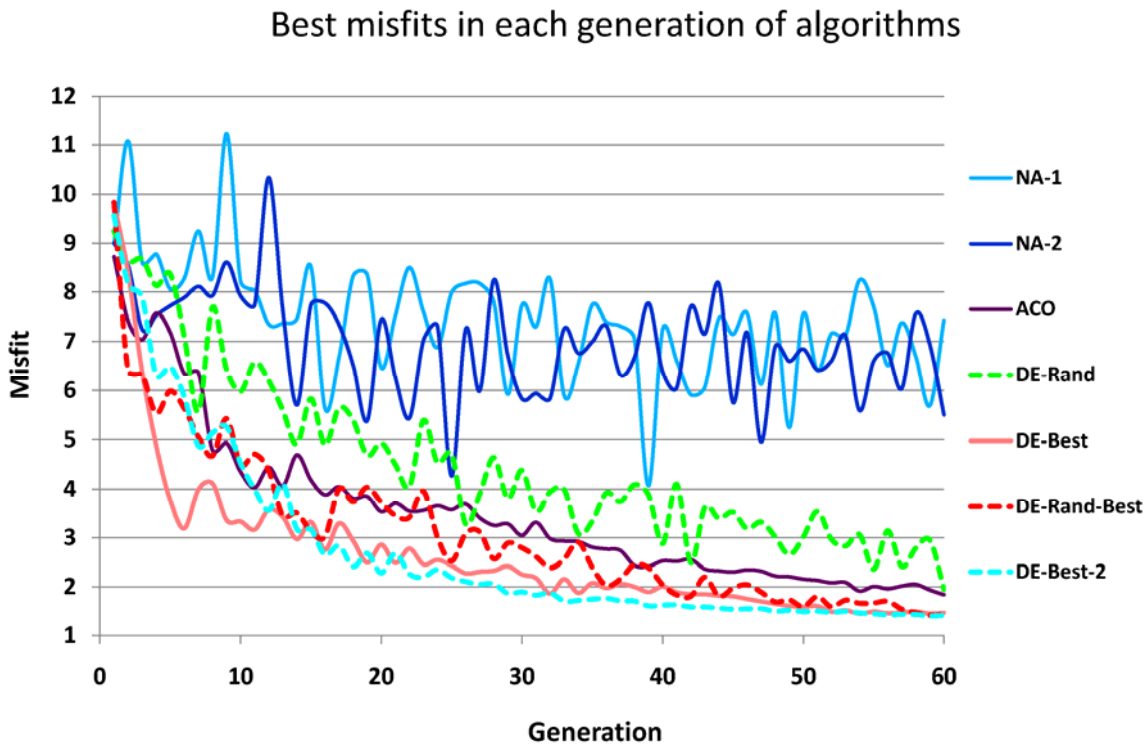

Figure 13: Comparison of convergence speeds for different algorithms in history matching of PUNQ-S3 model

As we can see in figure 13, DE-Best and DE-Best-2 have the fastest convergence in this problem. While DE-Best has a better performance in reducing the misfit value at initial stages of optimization (generations 1-15), DE-Best-2 takes the lead after generation 20. The DE-Rand-Best and ACO_R algorithms are ranked next in terms of convergence speed. Comparing DE strategies, the mechanism of base vector selection leaves its footprints on convergence speeds of the various strategies of this algorithm. DE-Best and DE-Best-2 have better convergence rates because the best member(s) (lowest misfit value) in the population is selected for adding the difference vector and building new candidate solutions. DE-Rand has more fluctuation in best generational value and a

slower convergence in comparison with the other three strategies of differential evolution. This can also be linked to the vector selection mechanism in the random strategy of differential evolution. The same concept, places the DE-Rand-Best convergence line between DE-Best-(2) and DE-Rand strategies. Both NA runs suffer from a poor convergence rate for this problem, probably due to the high number of unknowns. We should note that most of the algorithms (except NA runs) achieve their 50% efficiency in misfit reduction in the first 10 generations (500 simulations) by getting to misfit values less than 6. After this period, the effort of the algorithms is mostly devoted to find the global minimum and also discovering more local minima in the search space.

In figure 13, we see that the neighbourhood algorithm runs obtain only few models with relatively low misfits. Other algorithms exhibit a better performance by having a more stable reduction in misfit value and constantly finding good-fitting models. After generation 30, the best generational misfit values are less than 3 for DE-Best, DE-Best-2, DE-Rand-Best and ACO_R algorithms and less than 4 for the DE-Rand strategy. These models can result in potentially diverse realizations of the reservoirs with different rock properties. In fact, this is an example where we are able to produce multiple reservoir models with low misfit values. In the next section, we will investigate the sampling performance of algorithms in navigating the 45-dimensional space to understand if all of the good-fitting models found by different algorithms are actually satisfying this desired behavior.

It should be noted that conclusions drawn from this section might be specific to the selected setups and the fact that each of the simulations had been set to run once. A more robust conclusion would be obtained if the simulations were repeated many times using the same tuning parameters and different initial seed values.

5.2.7 Comparison of Sampling History

Another way to look at the history matching results from the PUNQ-S3 reservoir study is to compare sampling history figures for the various algorithms. These figures show how algorithms navigate through parameter space in each dimension. Figures 14-20 show the history trails for DE (Rand, Best, Rand-Best and Best-2 strategies), ACO_R and two NA runs. In each figure the variables of the best individuals (lowest misfit) in each

generation are plotted versus the generation number. Each figure has 45 tiles showing the 45 parameters we perturbed to obtain a match. We plot the scaled parameter values (0, 1) on the vertical axis where 0 and 1 are the minimum and maximum initial range of parameters. The generation number (0, 60) is plotted on the horizontal axis.

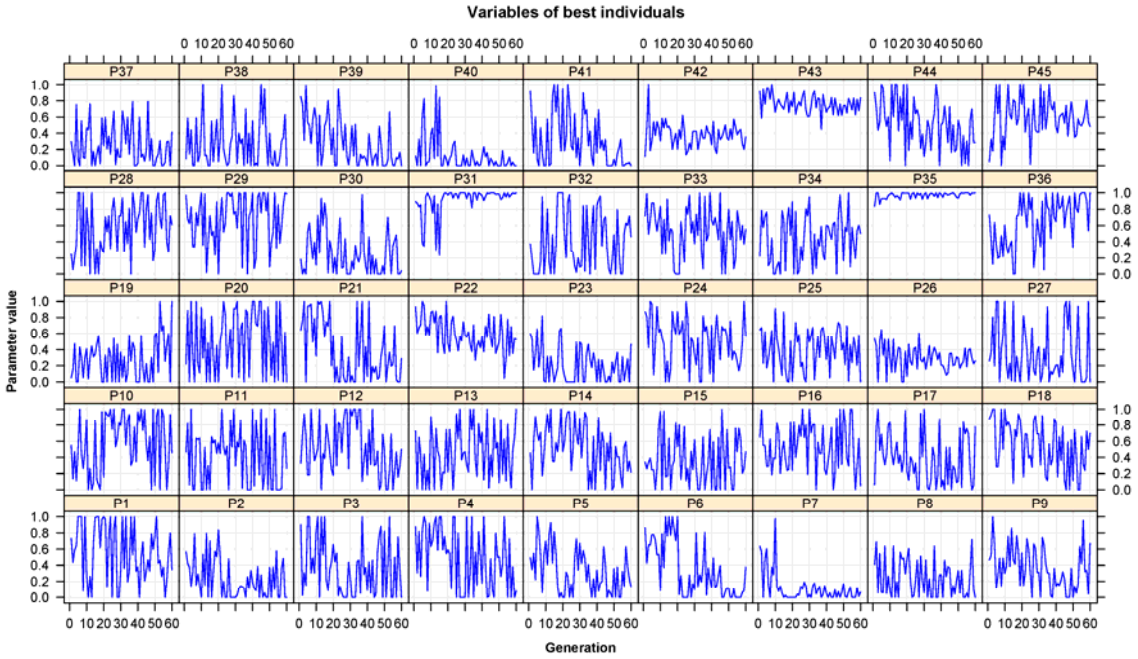

Figure 14: Sampling history of DE-Rand for PUNQ-S3 model

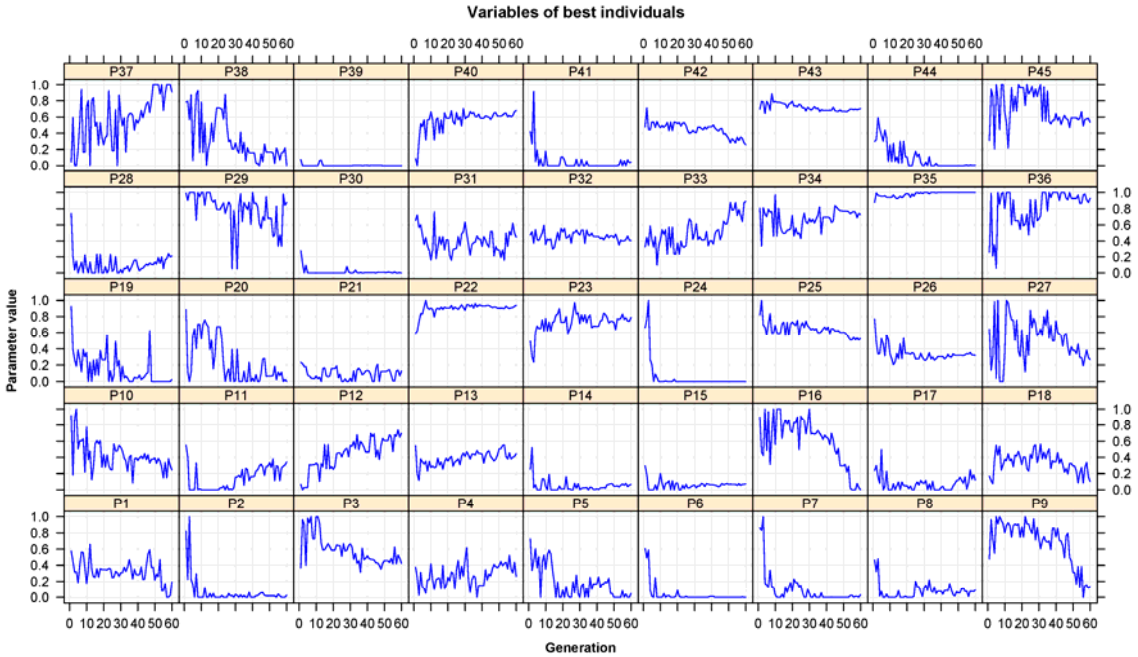

Figure 15: Sampling history of DE-Best for PUNQ-S3 model

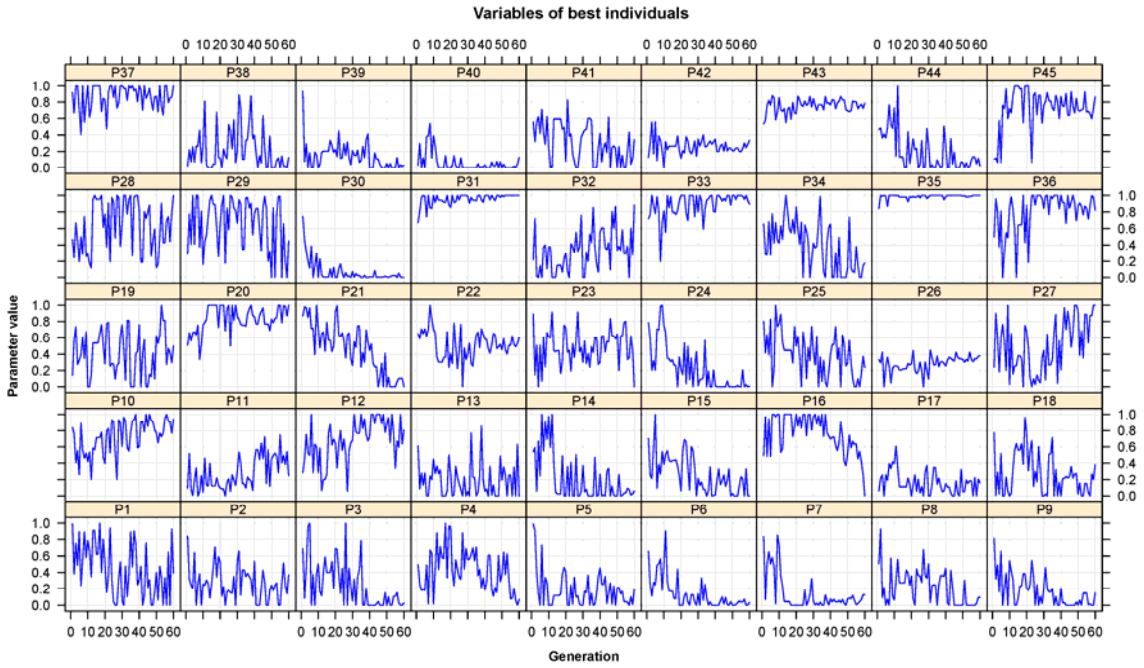

Figure 16: Sampling history of DE-Rand-Best for PUNQ-S3 model

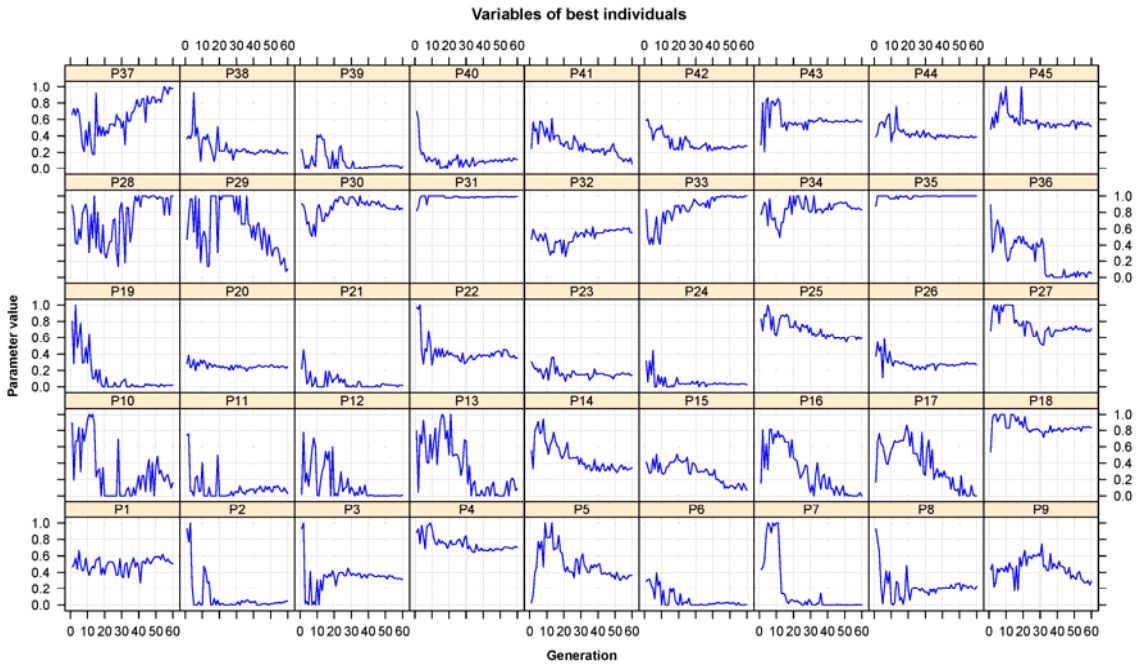

Figure 17: Sampling history of DE-Best-2 for PUNQ-S3 model

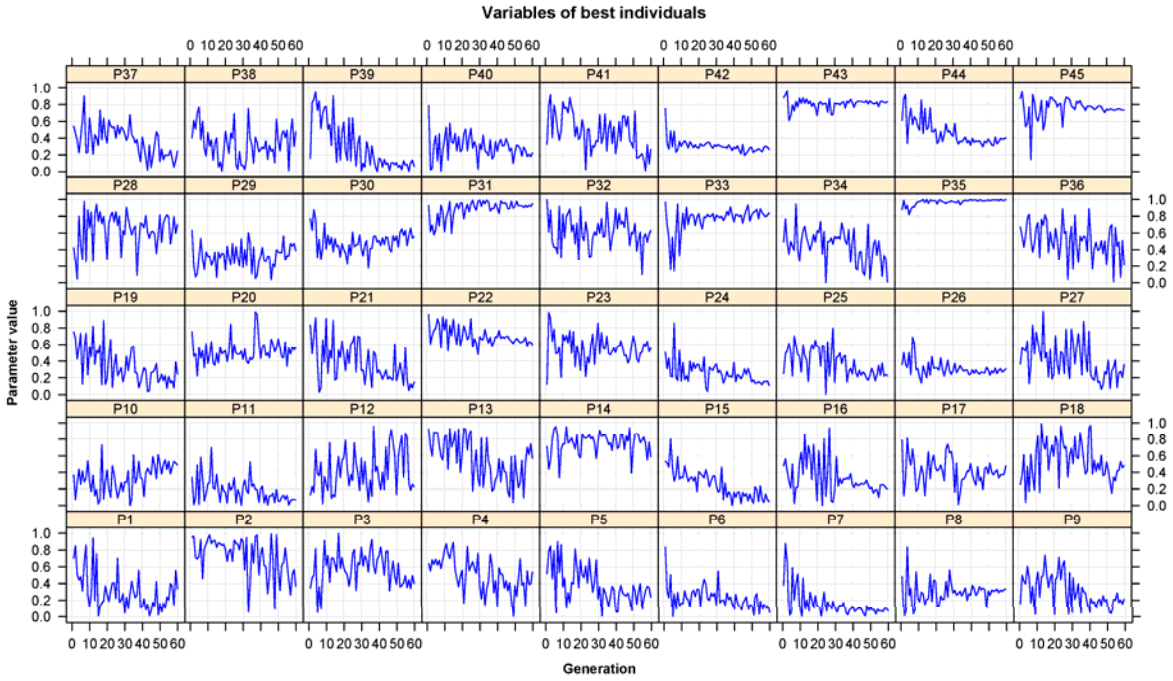

Figure 18: Sampling history of ACO_R for PUNQ-S3 model

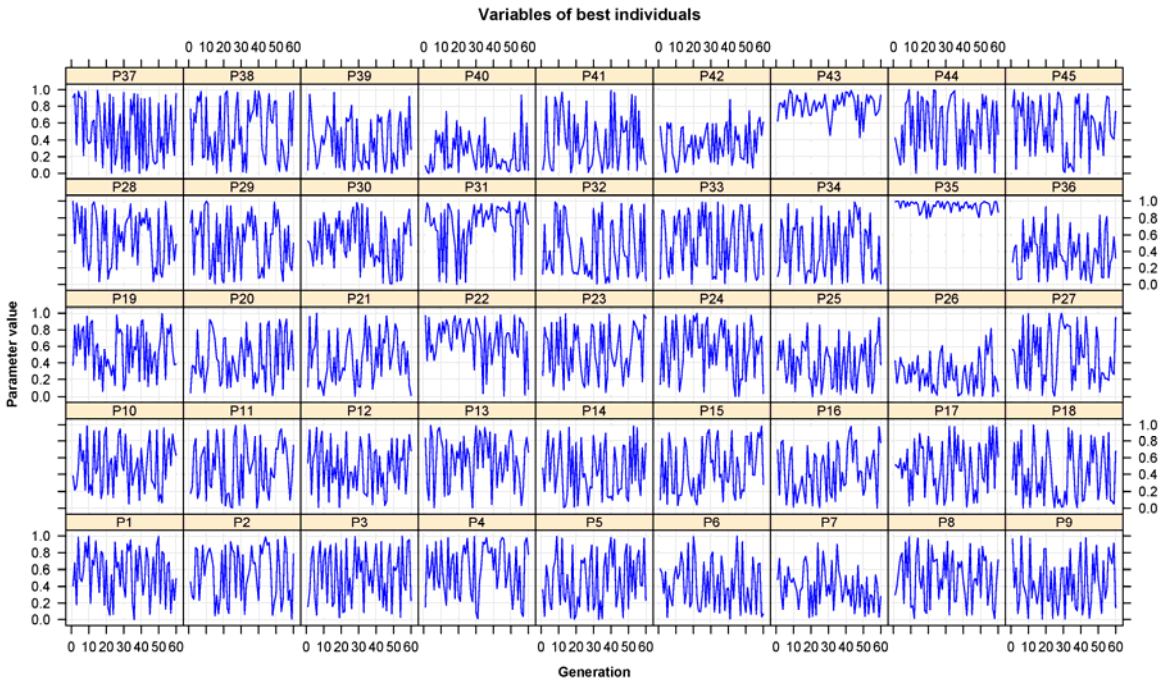

Figure 19: Sampling history of NA-1 for PUNQ-S3 model

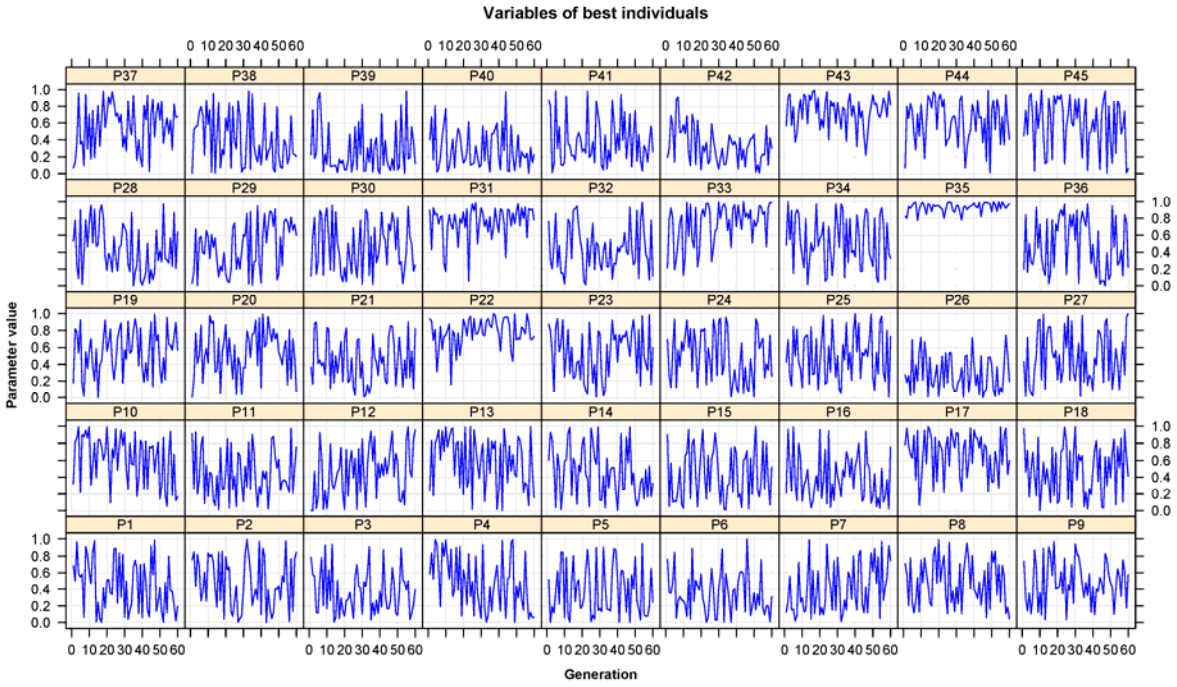

Figure 20: Sampling history of NA-2 for PUNQ-S3 model

The sampling history (figures 14-20) provide an insight into the performance of the algorithms. For example, in figures 14 and 15, we can compare two strategies of DE (Rand and Best). DE-Rand, due to its vector selection mechanism, has a wider sampling in parameter space, while in DE-Best less variation in the variables of best members are observed. We can see the effect of n_s and n_r tuning parameters on the performance of the neighbourhood algorithm (NA) in figures 19 and 20. NA-1 with higher exploration settings ($n_s = n_r$) has more fluctuation in sampling performance (for example in parameters 22, 31, 33, 37). Both NA runs cover most of the search space, but considering their convergence performance, these runs are not successful in locating the good fitting models in this problem.

In the sampling figures, we also notice that each algorithm focuses on different regions for the parameters in search space, resulting in diverse history-matched reservoir models. For example, DE-Rand-Best and DE-Best share an almost equal best misfit value, and looking at figure 13, their generational best misfit values for the 10 final generations exhibit a similar trend. However, these two strategies have a completely different behavior in their sampling. This difference comes in two areas: jumps of the best generational misfits and focus on different regions of the search space. For

example, the DE-Rand-Best strategy has more jumps in the last 10 generations in parameters 1, 23 and 27. This behaviour shows that diversity of the models (differences between rock properties) for the best fitting models is larger in DE-Rand-Best in comparison with DE-Best-2. The second difference between these two strategies is their concentration on distinct areas of the search space. A close look at the sampling history of parameters 34 and 44 in the final generations reveals that two different areas of the parameter space are under focus. Another example is the final misfit values of the DE-Rand and ACO_R algorithms which are quite close (1.83 and 1.95 respectively). Examining the location of parameters in the last generations of these algorithms (figures 14 and 18), we can see different final values, for example in parameters 4, 19, 20, 24, 29, 30, 34.

This diversity of the reservoir models, as demonstrated with the sampling history figures, explains the different match quality and predictions in figures 6-12 for the algorithms. These differences in reservoir parameters are likely to influence the oil recovery estimates after 16.5 years in the PUNQ-S3 reservoir. We should note that these differences are observed between the best history-matched models for each algorithm. Other models in the ensemble of 3000 solutions may also have similar low misfits, but different combinations of reservoir parameters. The algorithms used in this study converge to different regions of the search space while having low misfit values. Different regions of the search space, essentially, mean different realizations of the reservoir resulting in a good match. In section 3, we will explore the uncertainty of the predictions made by the ensemble of history matched-models and examine the effect of sampling history on the resulting uncertainty envelope.

5.3 Uncertainty Quantification

Partners of the PUNQ project, beside parameterization and history matching, needed to face another big challenge – prediction of recovery from this reservoir after 16.5 years. Table 16 summarizes the different methods used for history matching and uncertainty quantification in the PUNQ-S3 project by each partner of the original project.

Table 16: Different methods used in PUNQ-S3 project for HM and UQ [Boss, 1999]

Approach	TNO-1	TNO-2	TNO-3	Amoco-Iso	Amoco-Aniso	Elf	IFP-STM	IFP-Oliver	NCC-Oliver	NCC-GA	NCC-AG-MCMC
Param. domains	Homog. layers	Homog. drainage area regions	Homog. flow path regions	Fixed pilot-points Isotropic variogram	Fixed pilot-points Aisotropic variogram	Pilot-points selected	Fixed pilot-points	Fixed pilot-points	Fixed pilot-points	Parameters acting on whole grid	Parameters acting on whole grid
Params.	PORO, Kv, Kh uncorr.	PORO, Kv, Kh uncorr.	PORO	PORO, Kv, Kh correlated	PORO, Kv, Kh correlated	PORO	PORO	PORO	PORO, Kv, Kh correlated	PORO, Kv, Kh correlated	PORO, Kv, Kh correlated
Spatial technique	Piecewise constant	Piecewise constant	Piecewise constant	GRF	GRF	GRF	Kriging	Kriging	GRF	GRF	GRF
Uncertainty Quant.	Oliver-full	Oliver-full	Oliver-full	Multi-ML + start from random prior models	Multi-ML + start from random prior models	Multi-ML + start from sampled prior models	Scenatio test method (ML+)	Pliver-Pro start from kriged solution	Oliver-full	Multi-ML	Posterior sampling
Optim. algorithm	Dog-leg + Broyden gradients	Dog-leg + Broyden gradients	Dog-leg + Broyden gradients	GA	GA	GN-SD hybrid + finite diff gradients	Gauss-Newton + simulator gradients	Gauss-Newton + simulator gradients	Gauss-Newton + simulator gradients	GA	GA
Simulator	ECLIPSE	ECLIPSE	ECLIPSE	ECLIPSE	ECLIPSE	ECLIPSE	ATHOS	ATHOS	ATHOS	MORE	MORE

In table 16, different abbreviations are used. For example, ML refers to the Maximum Likelihood model which is a single realization of reservoir obtained by minimizing the objective function while MAP is the Maximum A Posteriori model where a prior term is also included. Having obtained the ML/MAP solution, we can locally characterize the objective function around the selected "best" solution. This approach is called ML+ and MAP+ which benefit from local characterization of the likelihood function and posterior distribution for uncertainty quantification, respectively. Multi-ML and multi-MAP methods use different initial models in uncertainty quantification. In table 16, we also see the Oliver approach [Oliver et al. 1996] which combines the advantages of the above methods. The Oliver-Prod approach only uses samples from the production data and the Oliver-Full method uses samples from both the prior and the production data.

The methods used for history matching and uncertainty quantification in the PUNQ-S3 project are different in the following 3 aspects. The first difference is the parameterization of the reservoir model, the second is that they use different algorithms to find good matching models (gradient, stochastic) and the third is the approach they take to quantify the uncertainty. These three aspects are equally important in reservoir engineering studies. Prediction of total oil recovery after 16.5 years from the PUNQ-S3 model is one of the main challenges of this problem. It has been shown that many methods give a good history matching result, yet fail to correctly predict the total oil recovery after 16.5 years [Boss, 1999, Floris et al. 2001]. The choice of reservoir simulator also affects the uncertainty estimates. Boss [1999] concluded that using different simulators to model the same reservoir led to a systematic difference in production forecast in one of the three simulators (ECLIPSE, ATHOS and MORE). The assumptions in reservoir modeling and the choice of objective function may also influence the uncertainty estimates. Nevertheless, the ultimate goal of any history matching, regardless of the above choices, remains same – obtaining accurate and reliable forecasts.

For uncertainty quantification in the predictions of the PUNQ-S3 model, we follow the same procedure used in chapter 4 for the Teal South example. Our approach is based on the evaluation of the posterior probability distribution based on a Bayesian framework and using multiple history matched models. We make our predictions based on the resampled model generated by the NAB algorithm [Sambridge, 1999 b]. Figures 21 and

22 show the resampled models obtained by NAB routine for DE-Rand algorithm and their match results for BHP of well 1 and cumulative oil production of the PUNQ-S3 reservoir. Resampled models by NAB have lower misfit values and thus better fit to the observations. We use these models and their corresponding posterior probability values to quantify the uncertainty associated with the future predictions.

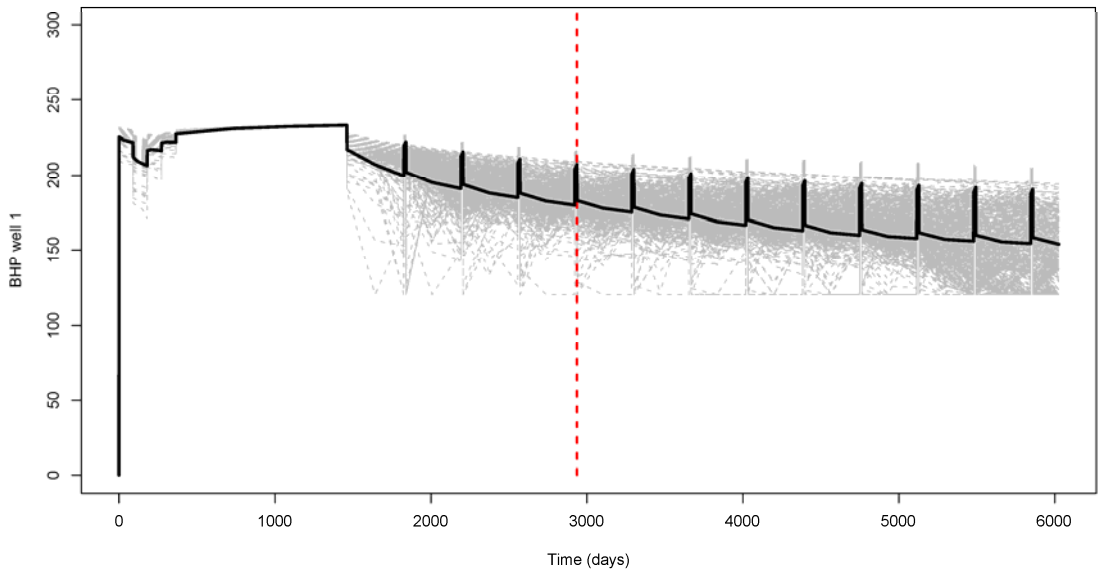

Figure 21: Resampled modes and their match results for BHP of well 1 using DE-Rand (vertical dashed line shows the end of history period, solid line represents truth case solution)

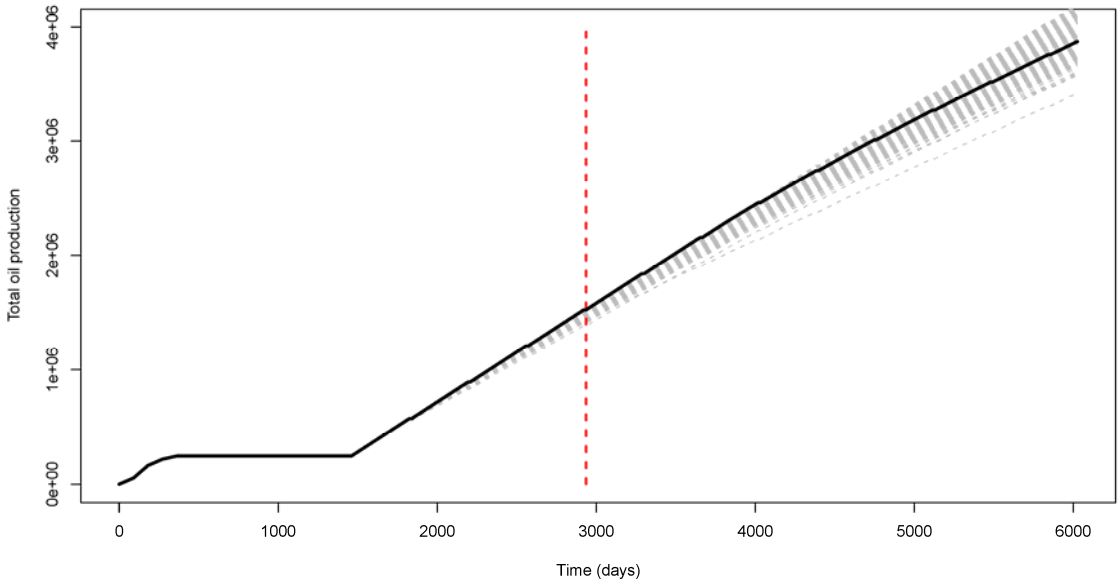

Figure 22: Resampled modes and their match results for total oil recovery from the field using DE-Rand (vertical dashed line shows the end of history period, solid line represents truth case solution)

In figures 23 and 24, we compare the results of published works for prediction of oil recovery in the PUNQ-S3 model and our work. Figure 23 shows this comparison between our work and the approaches that use a gradient-based optimization technique for finding good fitting models and figure 24 compares the stochastic-based methods used in previous publications and this work.

Gradient-based methods in this study tend to provide wider and less accurate credible intervals than stochastic algorithms, though some exceptions are possible. Also most of the stochastic algorithms achieve P50 prediction closer to the true solution. However, it is important not to underestimate uncertainty which may be the case when credible intervals are based on too few inferred models.

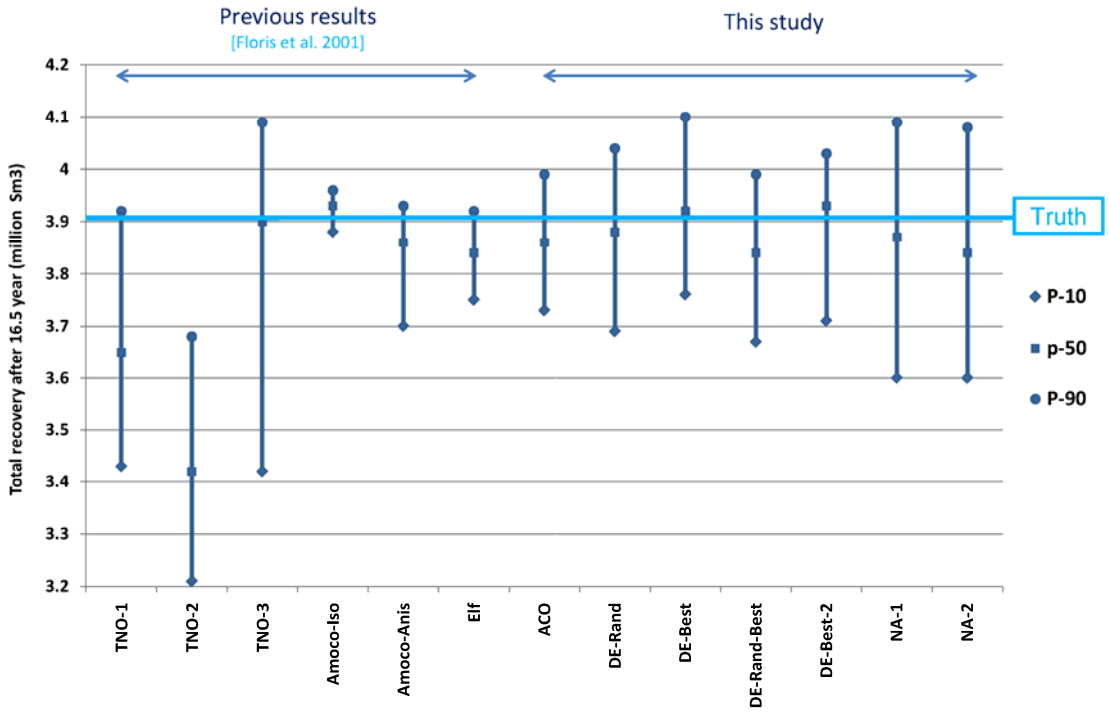

Figure 23: Uncertainty intervals in PUNQ-S3 model and its comparison with gradient methods and truth solution

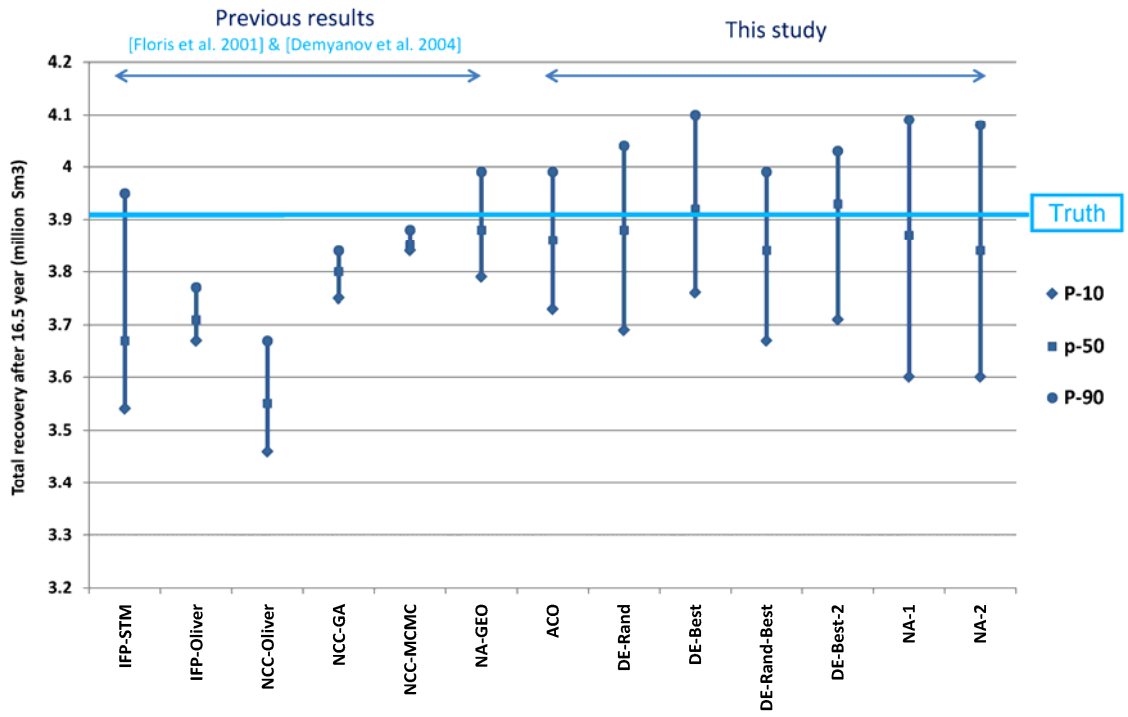

Figure 24: Uncertainty intervals in PUNQ-S3 model and its comparison with stochastic methods and truth solution

As we can see from figures 23 and 24, the uncertainty envelope of all the algorithms in this work comfortably cover the true oil production value from the PUNQ-S3 model. One interesting observation in these figures is the uncertainty range produced by the NA-1 and NA-2 ensembles. These two cases had a relatively poor performance in history matching (minimum misfits 4.07 and 4.26 respectively). It is interesting to note that the NA runs did not find very good fitting regions in the search space, but are able to make good predictions. Considering the NA-1 and NA-2 cases, NA-1 had a more exploratory setting and this is probably the reason for the closer-to-truth estimate of recovery using this setup. DE-Rand with its wide sampling and good misfit values obtains a very close P50 estimate to the truth solution. DE-Best and DE-Best-2 slightly over-estimate the oil recovery from the reservoir. This can be tied with their sampling history in figures 15 and 17. These two strategies have less variation in the variables of the best-misfit models and quickly zoom into final good-fitting regions in the parameter space. This behavior may explain their capability in making recovery estimates since they do not explore so many areas of the search space. Looking back to the sampling performance (figures 14 and 18), we can see that DE-Rand and ACO_R sample a wider range in parameter space in comparison with the DE-Best and DE-Best-2 strategies.

From these examples, we see that the performance of the algorithms in sampling of the search space is reflected in their ultimate recovery predictions.

Further to the above comparison, we have also run NAB routine on all trials of the differential evolution with four strategies. Table 17 presents the mean value obtained for P10, P50 and P90 estimate based on 10 trials in each strategy of the differential evolution algorithm. The tuning parameters for these runs were previously reported in tables 10-13.

Table 17: Mean value for the Bayesian credible intervals provided by different strategies of differential evolution in 10 trials

Strategy	P10	P50	P90
Rand	3.68	3.88	4.06
Best	3.70	3.89	4.05
Rand-Best	3.69	3.84	3.97
Best-2	3.66	3.85	4.04

While it is difficult to draw a definite conclusion (and hence provide an advice on which algorithm or strategy is the best option for obtaining accurate recovery estimate), we may propose a general conclusion based the uncertainty estimates in figures 23 and 24 and table 17. In PUNQ-S3 case, a more diverse sampling of the search space (DE-Rand and NA) leads to a recovery estimate which is very close to truth value. NA did not find very good fitting regions in the search space, but it was able to make good predictions with a wide sampling of the search space. On the other hand, we should also consider the computational resources available to perform history matching studies. A diverse sampling of the search space, usually, will translate to a slow convergence. From this point of view, and only comparing DE-Rand with NA, we see that DE-Rand is a better option. Comparing the convergence behavior of these two algorithms in figure 13, we see that DE-Rand enjoys a much faster convergence in comparison with NA. Since both algorithms provide a good uncertainty estimate, the DE-Rand strategy is the best choice of these two options.

Here we put forward another important question. Considering a limited number of simulations for any algorithm, how many simulations are needed to obtain a stable uncertainty estimate? Based on the results of this chapter, we know that the sampling performance of an algorithm affects the uncertainty estimate. Thus, uncertainty estimates obtained from an ensemble of 3000 simulations will probably be different if we only use the first ten or hundred members of the ensemble in our calculations. Theoretically, uncertainty estimates will vary during initial stages of optimization (sampling). In the history matching process, there should be a critical point where we have enough information about the topology of the search space to obtain a reliable uncertainty estimate. This form of comparison between history matching algorithms may be a better way for making judgments about their efficiency: a better algorithm will obtain a stable uncertainty estimate from fewer simulations. This issue will be discussed in the next chapter.

5.4 Concluding Remarks

This chapter presents a comparative study of recently-developed agent-based optimization algorithms for tackling history matching and uncertainty quantification problem in a high dimensional example. The results of this chapter provide some useful insights into performance of these algorithms in a challenging problem and the ties between sampling efficiency of the search space and uncertainty estimate of future reservoir behavior.

Since the main goal of this chapter is a comparative study of agent-based algorithms for history matching and uncertainty quantification, we have not focused on parameterization of the model. Thus, we have used a simple homogenous parameterization scheme in this work. Obviously, the proposed algorithms are not limited by the parameterization method and the choice we made here does not stop the user from trying advanced schemes such as geostatistical frameworks for reservoir parameterization.

As in one of the conclusions of the PUNQ project, Boss [1999] and Floris et al. [2001] stated that using porosity and permeability multipliers for homogenous layers or regions results in poor quality history matches and significant bias in the estimate of recovery

from truth value. We have shown that homogenous layer parameterization, as is still widely used in the industry, is a good option in preparing the reservoir model for history matching and can result in good-quality history-matched models.

For the Teal South reservoir, differences between the performance of the algorithms studied were marginal due to a relatively simple model with just eight free parameters and the univariate objective function based on the data from a single-well. In the PUNQ-S3 high dimensional problem, the gap in performance of algorithms increases. The neighbourhood algorithm performance is affected by the high number of unknowns in the problem and the algorithm fails to locate good-fitting regions of the search space.

Different algorithms are able to find multiple models of similar history match quality located in different regions of the parameter space. Such diverse models are essential for adequate representation of uncertainty.

All algorithms provide the confidence bounds which comfortably includes the true solution. The P50 prediction is close enough to the true solution in particular for DE-Rand and NA. The sampling behavior of the optimization algorithm has a direct impact on its prediction. Thus, the algorithms performing more exploratively tend to provide wider prediction confidence bounds.

Chapter 6

> "I can write better than anybody who can write faster, and I can write faster than anybody who can write better"
>
> A. J. Liebling (1904-1963)

Multiobjective History Matching and Uncertainty Quantification

In chapters 4 and 5 we applied ant colony and differential evolution algorithms to the history matching and uncertainty quantification of two reservoir models. In these models we either had a single objective (matching to oil rate only, in Teal South model) or multiple objectives where we summed all misfit values to form a global objective function (matching to bottomhole pressure, gas oil ratio and water cut of different wells in the PUNQ-S3 model). This chapter talks about a novel way of handling different objectives in history matching using a multiobjective optimization approach.

This chapter consists of three main sections. In the first part, we introduce the concept of multiobjective optimization and different ways to solve this class of optimization problem. In the second part we describe our approach for extending the current single-objective versions of ant colony optimization and differential evolution algorithms for solving multiobjective problems and test these algorithms for benchmark functions. The final section of this chapter will focus on the application of the proposed methodology to history match the PUNQ-S3 model. This part covers the comparison of single and multiobjective optimization algorithms for history matching and uncertainty quantification.

6.1 Multiobjective Optimization

Multiobjective optimization is synonymous with "multivector optimization", "multicriteria optimization", or "multiperformance optimization". Multiobjective optimization dates back almost 130 years. Economics professor Francis Edgeworth was the first to propose a multiobjective economic decision making framework [Edgeworth, 1881]. Vilfredo Pareto later created his important theory called *The Pareto Optimum*. Pareto [1906] states: "The optimum allocation of the resources of a society is not attained so long as it is possible to make at least one individual better off in his own estimation while keeping others as well off as before in their own estimation". Based on these ideas frameworks of multiobjective optimization were developed.

In single objective optimization we consider minimizing or maximizing only one objective, without considering the effect of other criteria. As an example, figure 1 shows a water flooding project for increased oil recovery. For a single objective optimization, the goal is to maximize oil recovery. However a better solution only in terms of the oil recovery may come at the cost of increased cumulative water injection. This will increase the cost of water flooding and the chance of early water breakthrough in production wells. If increased oil production was the only objective for this problem, solution 3 would be the best option for a decision maker. In such a case all water flooding projects will operate with maximum possible water injection. However due to technical and economical constraints, we should also consider the water injection as the second objective in this problem. This implies a control over water injection in order to maximize the breakthrough time and sweep efficiency while optimizing the injected water. Also considering some technical limits, it might not be possible to have operations at specific intermediate rates which adds another challenge in this multiobjective optimization problem.

In this example, all technical and economical constraints including oil production and water injection should be optimized in order to have a successful project. This is a very simple illustration of multiobjective problems that petroleum engineers may face in field operations. These challenges usually deal with multiple conflicting objectives that should be considered simultaneously in order to make decisions.

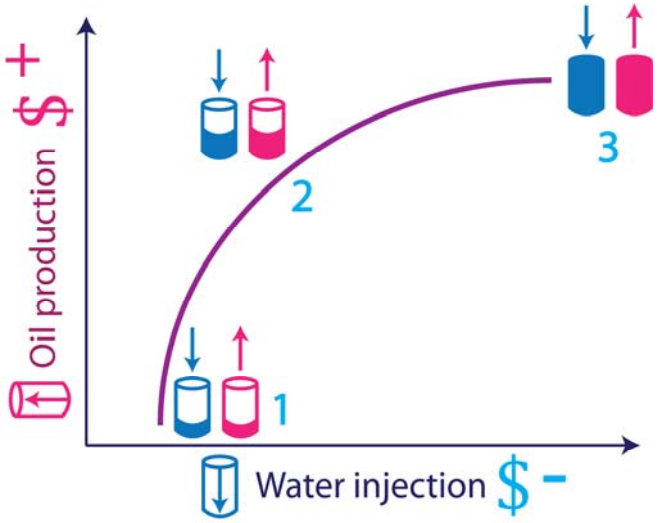

Figure 1: Water injection vs. oil production from a reservoir

6.1.1 Single and Multiobjective Optimization

Multiobjective optimization is defined as the task of finding one or more optimum solutions when a problem involves more than one objective or goal. Unlike single objective that deals with a single space (decision variable space), in multiobjective optimization, a new space is also considered which is called the objective function space. Figure 2 shows the decision variable space (where we search the parameters) and objective function space (where the solutions are evaluated) for a bi-objective optimization problem with three decision variables. This follows the waterflooding project introduced in the beginning of this chapter.

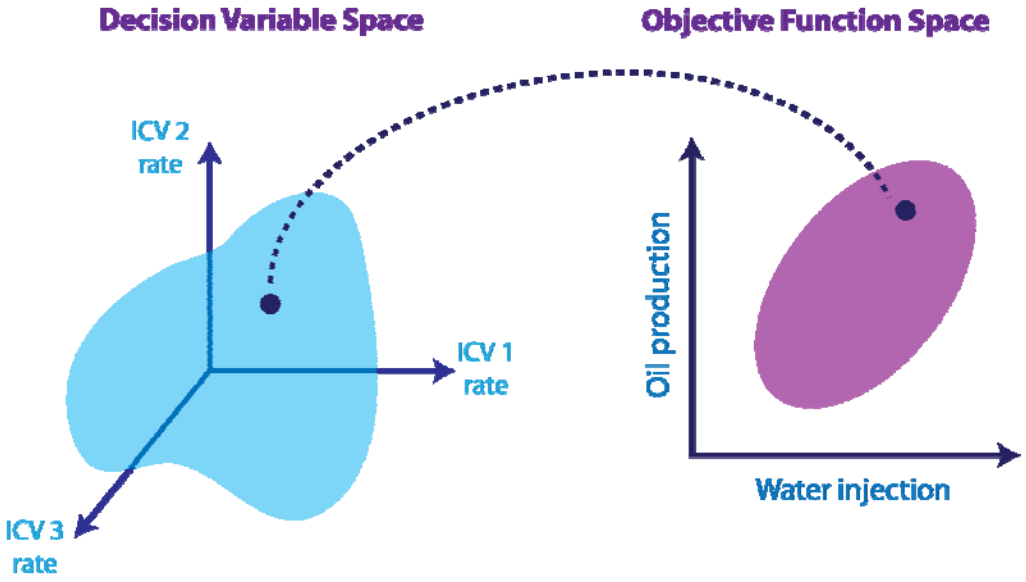

Figure 2: Decision variable and objective function spaces for the water flooding example

In this project, water is injected through a single well which is equipped with three intelligent control valves (ICV). These valves control the injection rate of water in different segments of the well. Their rates form the three decision variables for this problem. Any combination of injection rates in the decision space (ICV settings) can be mapped to the objective function space (resulting water injection and oil production).

In a multiobjective optimization problem, the decision vector is denoted by $\boldsymbol{x}$ and the decision space is shown by X. Similarly, the objective vector is denoted by $\boldsymbol{y}$ and Y represents the objective space. The multiobjective optimization, in general form, can be formulated as:

$$\left.\begin{aligned} &\text{Maximize/Minimize } f_m(\boldsymbol{x}), && m = 1,2,\ldots,M \\ &\text{Subject to} \quad g_j(\boldsymbol{x}) \geq 0, && j = 1,2,\ldots,J \\ &\qquad\qquad h_k(\boldsymbol{x}) = 0, && k = 1,2,\ldots,K \end{aligned}\right\} \quad (1)$$

where solution $\boldsymbol{x}$ is a decision vector of n variables: $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$. M is the number of objective functions in the problem which can be minimized or maximized: $f(\boldsymbol{x}) = (f_1(x), f_2(x), \ldots, f_M(x))$. The multiobjective optimization problem also may have constraint functions ($g_j(\boldsymbol{x})$ and $h_k(\boldsymbol{x})$) which determine the set of feasible solutions.

6.1.2 Pareto Dominance Concept in Multiobjective Optimization

In any single objective optimization, the optimal solution is the one that gives maximum (or minimum) value of the objective function. However, in the context of a multiobjective optimization, the notion of optimality is different and we are interested in finding good compromises among the objectives that we wish to optimize. In multiobjective problems, we need to define the concept of dominance. In these problems, among the several possible solutions, moving from one of the solutions to another, causes an improvement for that objective function, while making the other one worse. This notion of optimality is called Edgeworth-Pareto optimality, or in the most commonly used form, Pareto optimality or dominance.

In general terms, dominance implies that if there exists an alternative solution (A) that is at least equal to (B) in terms of all objective functions, and if A is strictly better than B for at least one of the objective functions, then A dominates B (A $\underline{p}$ B). This can be expressed as following two conditions:

1) f_m (**A**) $\succ$ f_m (**B**) for all m = 1,2, …, M (A is no worse than B for all objectives) AND 2) f_m (**A**)$<$$f_m$ (**B**) for at least one m = 1,2, …, M (A is better than B for at least one objective)	(2)

Figure 3 shows a graphical representation of the Pareto dominance concept for a two-objective problem where both objectives are minimized. Six solutions for this bi-objective problem have been generated and are shown with dots on this figure.

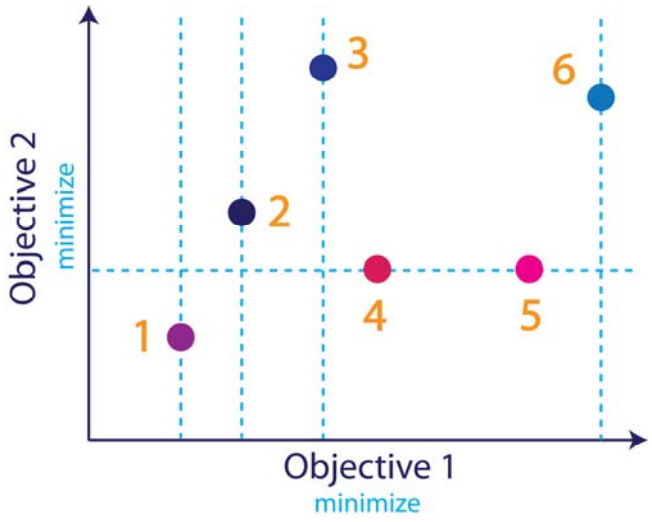

Figure 3: An example of a bi-objective optimization problem with six solutions

For example, let's compare solutions 1 and 2. Solution 1 is better (minimized) for both objectives 1 and 2. We can say that both Pareto dominance conditions are true and solution 1 dominates solution 2. Similar comparison can be made for solutions 4 and 5. Solution 4 is equal to solution 5 for objective 2 and it is better for objective 1, thus solution 4 dominates solution 5 in this example.

A solution is called Pareto optimal if there exist no feasible vector of decision variables which would decrease some criterion without causing a simultaneous increase in at least one other criterion. In other words, any of solutions on the Pareto front can not be improved without causing degradation in at least one other objective of the optimization problem.

We follow two main objectives in solving any multiobjective optimization:

1) Obtain an ensemble of solutions as close as possible to the true Pareto front
2) Obtain an ensemble of solutions as diverse as possible on the Pareto front

The Pareto dominance concept is used to evaluate and compare the generated solutions in most of the algorithms designed for multiobjective optimization. In these methods more weight is given to the solutions that are not dominated during the optimization (for example solution 1 in figure 3). We will describe this class of methods in the next section. The proposed extensions for ant colony and differential evolution algorithms for tackling multiobjective history matching problems are also based on the Pareto optimality concept.

6.1.3 Different Ways to Handle Multiple Objectives

Several strategies are used in the optimization community to handle multiple objectives. We present the classification proposed by Cohan and Marks [1975] which identifies three main groups of approaches for tackling multiobjective optimization problems, a priori, progressive and a posteriori methods.

6.1.3.1 A Priori Articulation of Preferences

This class of methods for solving multiobjective optimization problems are based on a "decide and then search" scheme. Based on the relative importance of objectives, a priori methods allow the user to specify preferences before staring the optimization process.

6.1.3.1.1 Weighted Sum Approach

This method is the simplest and most common way of optimizing two or more variables. In this approach, as the name suggests, the objective functions of the problem

are scalarized into a single objective by pre-multiplying each objective with a weight factor. For a bi-objective optimization problem, this can be formulated as:

$$Obj_{Global} = (w_1 \times Obj_1) + (w_2 \times Obj_2) \quad (3)$$

There are several problems with the weighted sum approach. The first question that should be addressed is the choice of the weights themselves. The answer to this question is not easy and depends on the importance of each objective and the decision of the user to assign weights to these objectives. The weights also depend on the scaling of each objective function. Different objectives may have different orders of magnitude. Thus, there is a need for normalization of the objective function values. Determining the correct weights is one of the major difficulties in this approach. The second drawback of the weighted sum method is related to its inability to cover non-convex Pareto fronts [Das and Dennis, 1997]. Even for optimization problems with convex Pareto front, the weighted sum approach may miss some parts of the front and often the optimal solution distribution is not uniform. In these cases, by choosing different weights, one at a time, different regions of the Pareto front can be discovered. Since there exists little or no information about the shape of the true Pareto front (PF_{true}) in real world applications, the blind use of the weighted sum approach may result in missing some parts of the Pareto front and the opportunity to find all possible solutions.

6.1.3.1.2 Fuzzy Logic Approaches

Fuzzy logic provides a tool to compute with words in an uncertain and vague environment. In fuzzy logic there is no sharp limit between numbers; instead a membership function is defined to express the degree of transition between numbers. Membership (μ) states the degree of belongness to the special set: $\mu = 0$ means that a solution does not belong to the specific set and $\mu = 1$ puts the selected solution completely in the desired set.

The concept of fuzzy numbers and logic can be used in multiobjective optimization. In a multiobjective problem, one can assign a degree of satisfaction for each objective using the membership function. The value of that particular objective function is fuzzified by μ to give a value in the range $\{0,1\}$, which is a measure of how well a solution satisfies the requirement. Figure 4 shows an example of fuzzy objectives for the water flooding

example in the beginning of this chapter. In this water flooding project, instead of exact numbers for water injection and oil production as the objectives of the problem, we deal with fuzzified information.

In this example, fuzzified numbers (logical values) of water injection and oil production contribute to understanding the balance between these two objectives and optimizing the recovery process.

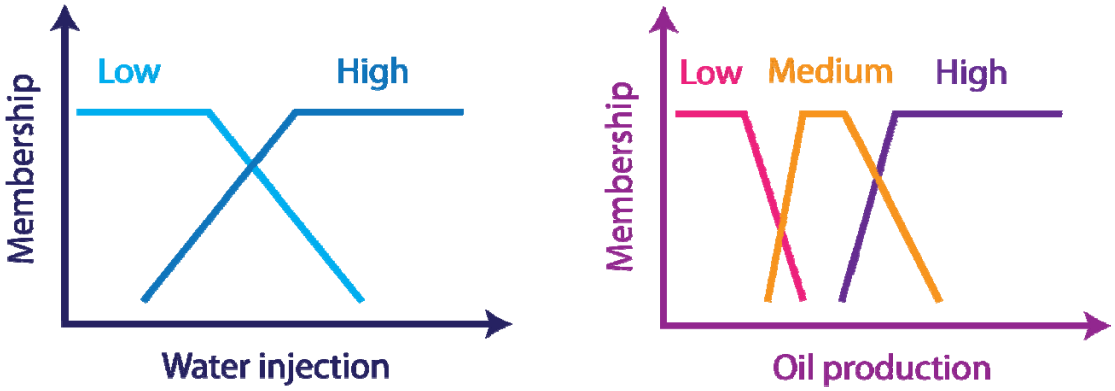

Figure 4: Fuzzy membership functions for water injection and oil production

6.1.3.1.3 Lexicographic Methods

In lexicographic methods, objectives are ranked based on their relative importance for the decision maker. The lexicographic approaches assume the decision maker has a strictly ordered preemptive preference system among objectives with fixed target levels. Objective functions are prioritized based on the user's preference. In this framework, there are some priority levels for the optimization problem. Minimization of a deviation in a higher priority level objective function is more important than other low priority levels. We should note that lexicographic approaches are suitable only when the importance of each objective is clearly known. These techniques explore the objective space unequally because priority is given to solutions performing well in one objective over another(s).

6.1.3.1.4 Adaptive Scalarization Methods

Recently there has been an increase in the number of the approaches which adaptively select the parameters used for scalarization of the objective functions. For example Eichfelder [2008] proposed an adaptive control approach based on sensitivity analysis of the objectives. Other examples in this category include Kim and de Weck [2005], Li

et al. [2009] and Ryu et al. [2009]. These methods aim to successfully handle non-convex fronts in multiobjective optimization.

6.1.3.2 Progressive Methods

Progressive methods follow the "search and then decide" pathway to find Pareto optimal solutions. Progressive or interactive methods work with simultaneous update of the preference of objectives. Decision maker and optimization toolbox are intertwined in this approach. A problem may be too complex to specify a preference for objectives before starting the optimization run (a priori methods). However as the search moves on, based on the new results, one may be able to give a specific preference to objectives. This forms the framework of interactive methods in tackling multiobjective optimization problems. Miettinen et al. [2008] identified three main groups in the progressive techniques: trade-off based methods, reference point approaches and classified methods.

In trade-off based methods usually at each iteration either a trade-off between objectives are presented to the decision maker or the decision maker is asked to input subjective trade-off. For example in the Zionts-Wallenius or Z-W method [Zionts and Wallenius, 1976] or the Interactive Surrogate Worth Tradeoff (ISWT) algorithm [Chankong and Haimes, 1983] several trade-offs are shown to the decision maker. In the Geoffrion-Dyer-Feinberg (GDF) approach [Geoffrion et al. 1972], Sequential Proxy Optimization (SPOT) method [Sakawa, 1982] and GRadient based Interactive Step Trade-off (GRIST) algorithm [Yang, 1999] the decision maker controls the search direction in each iteration by providing his preference.

Reference point approaches work by the decision maker specifying aspiration levels for all objective functions. Then the system optimizes the objective functions and in the next step the decision maker is free to adjust the reference points already specified at the first stage. Examples in this group include works by Wierzbicki et al. [2000] and Steuer [1986].

In classification-based methods the objective functions are classified by the decision maker to direct the optimization towards his/her preferences. Based on the Pareto optimality concept, it is not possible to improve a Pareto optimal solution in one

objective, unless we make the other objective worse. This concept is used in classification-based methods where the decision maker tells the system which objective functions at each iteration should be optimized and which ones should be compromised. The step approach (STEM) [Benayoun et al. 1971] and Nondifferentiable Interactive Multiobjective Bundle-based Optimization System (NIMBUS) methods [Miettinen, 1999] are two examples in this group.

In all of the progressive methods the role of the decision maker (DM) is very important. This critical interaction helps to improve the quality of the solutions. However there is less effort to propose and implement progressive multiobjective optimization algorithms in real world engineering applications. This might be due to the additional effort and willingness required by DM to provide interaction. Complexity of the problems in this area might be another barrier to apply these methods in challenging real life problems where the decision maker is unable to provide his preference even after getting some solutions during the optimization.

6.1.3.3 A Posteriori Aggregation of Preference

A posteriori methods for solving multiobjective optimization problems work on the "search and then decide" principle. In these approaches, an ensemble of solutions is presented to the decision maker and he plays his role by reviewing and selecting the appropriate individuals after finding multiple optimal solutions. Generally, we can recognize two main groups in a posteriori methods: multiobjective population-based and evolutionary algorithms and non-evolutionary methods such as the e-constraint approach.

Stochastic evolutionary and population-based methods are one of the three fastest growing fields of research and application among all computational intelligence topics [Deb, 2008]. These algorithms work with a population of individuals and provide an opportunity to generate an ensemble of diverse and evenly distributed solutions on the Pareto front. After generating an ensemble of Pareto optimal solutions, the user can select some of the individuals from this ensemble using some high level information. This feature makes evolutionary and stochastic agent-based methods an excellent candidate for solving multiobjective problems.

A complete review of evolutionary algorithms for multiobjective optimization can be found in Deb [2001] which provides description, hand calculations, advantages and disadvantages of each algorithm. In this section we review non-elitist and elitist-based methods and show some examples in each section.

6.1.3.3.1 Non-Elitist Methods

In this group of methods, the evolutionary algorithms are modified to handle multiple objectives; however they do not employ an elite-preserving mechanism. As the first real implementation of a multiobjective evolutionary optimization algorithm, The Vector Evaluated Genetic Algorithm (VEGA) was proposed by Schaffer [1984]. It uses subpopulations that optimize each objective of the problem separately (independent selection cycles regarding each objective). VEGA does not incorporate the concept of Pareto optimality in the selection mechanism of GA. Later Goldberg [1989] proposed a better implementation of dominance concept by performing a "nondominated sorting". Goldberg suggested using the dominance score as a metric for the survival of individuals in genetic algorithm. In this framework, nondominated solutions will have more chance to proceed to next generation of the algorithm. Goldberg also suggested using niching methods to increase the diversity of the solutions on Pareto front. Niching is an operator to promote diversity of the solutions by controlling the selection pressure of population members. It aims not to allow a single individual to take over the population. Different niching techniques are used in the literature for evolutionary algorithms with the *sharing* approach being a famous and widely used method.

The Multi-Objective Genetic Algorithm (MOGA) was developed by Fonseca and Fleming [1993]. In MOGA each individual gets a rank corresponding to the number of individuals in a current population by which it is dominated. All nondominated solutions are assigned a rank equal to 1. All other solutions receive a rank by calculating the number of solutions (n) that dominate the particular individual, thus assigning rank (n+1) to it. In this way, there will be at least one solution with rank 1 in the population and the maximum rank of any individual solution will be N (population size). In MOGA, lower rank solutions (dominated by less members) form the mating pool in the genetic algorithm. MOGA uses niching and fitness sharing among solutions of each rank to maintain solution diversity.

The Nondominated Sorting Genetic Algorithm (NSGA) was proposed by Srinivas and Deb [1994]. NSGA employs several layers of classification for individuals by sorting the population based on the Pareto dominance concept. In this algorithm, after the first non-dominated class of solutions are identified, a dummy fitness value is assigned to them. Then this class is temporarily discarded and a second layer of solutions receive their fitness values. This procedure is continued until all population members receive their scores. NSGA also employs a fitness sharing mechanism to ensure the diversity of the solution.

MOGA and NSGA use the same concept in selecting the individuals for genetic algorithm operations. They only slightly differ in the way dominance ranks are assigned to individuals in each generation. Figure 5 illustrates their conceptual difference in assigning dominance scores to population members.

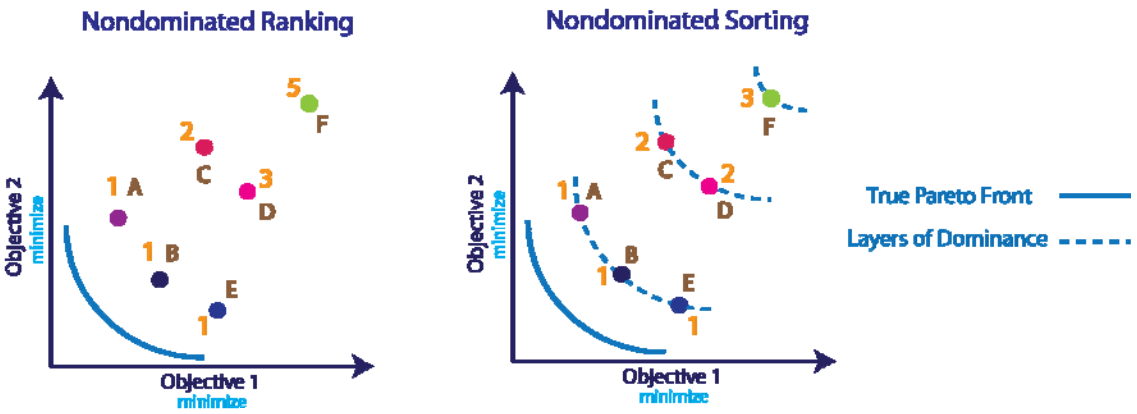

Figure 5: Illustration of Pareto ranking (MOGA) and Pareto sorting (NSGA)

NSGA uses nondominated sorting where several layers (fronts) of dominance are identified. MOGA employs nondominated ranking which ranks the population according to the 'degree of dominance'. This ranking is performed by counting the number of individuals that dominate each member of the population. The more members of current population dominate each individual, the higher its dominance score and the lower its survival chance to next generation.

Non-elitist based methods suffered from the disadvantage that elite members in each generation were not systematically preserved for later generations. To overcome this

drawback, a major research front was initiated to incorporate elite preserving mechanisms in operations of multiobjective optimization algorithms.

6.1.3.3.2 Elitist-Based Algorithms

Elitist-based multiobjective optimization methods were introduced as the second generation of evolutionary and population-based algorithms to overcome the disadvantages of non-elitist based approaches. Elitist-based algorithms use special operators to favor survival of the elite members of the population to the next generation.

The Strength Pareto Evolutionary Algorithm (SPEA) was introduced by Zitzler and Thiele [1999]. It uses an external archive containing nondominated solutions previously found. A combined pool of current and external archive is first formed at each iteration. SPEA computes a strength value similar to the ranking value used by MOGA. This value is proportional to the number of individuals that dominate the selected member and is used to assign a fitness value to it. SPEA also uses a clustering technique to maintain diversity among the solutions and to keep the archive size fixed. In the SPEA algorithm the size of the external archive is important and therefore should be selected carefully to ensure a successful optimization attempt.

Pareto-Archived Evolutionary Strategy (PAES) [Knowles and Corne, 2000] is another example of elitist-based multiobjective optimization algorithms. PAES also maintains an external archive of promising solutions. This algorithm compares the offspring with the parent and if the child dominates the parent, the offspring is accepted as the next parent. If the parent dominates the offspring solution, the offspring is deleted and a new solution is generated. If the parent and offspring do not dominate each other a crowding procedure is used to make the choice of the individual proceeding to next generation. In each generation, the offspring solution is compared with the archive. Only if the offspring solution dominates any member of the archive it is accepted as the parent and replaces the dominated solution.

Deb et al. [2002] proposed the elitist-based NSGA-II algorithm which turned out to be one of the most popular and widely-used algorithms to solve multiobjective optimization problems. In the NSGA-II algorithm, the offspring (size N) and parent populations (size N) are combined at each generation to form a population of size $2N$.

Then this joint population is sorted and classified into different fronts of Pareto optimality (like NSGA). The next generation of solutions is chosen from the pool of combined population (size 2*N*). The priority is given to the first class (front) individuals to fill the available slots in the new population. Since there are only *N* available slots in the new population, all of the combined pool individuals can not be accommodated in the new population. When it comes to fill the last available slots, there may exist more individuals than the remaining positions. In this case, the crowding-sorting operation comes to play and less crowded solutions are placed in the new population.

Other examples of stochastic population-based and evolutionary algorithms for multiobjective optimization include: cultural algorithm [Coello and Landa, 2003], Multi-Objective Messy GA (MOMGA) [Veldhuizen and Lamont, 2000], micro-GA [Coello and Toscano, 2000], Go With the Winner (MOGWW) [Brizuela and Gutierrez, 2005], estimation of distribution algorithms [Pena et al. 2005], particle swarm optimization [Alvarez-Benitez et al. 2005] [Mostaghim and Teich, 2003] and artificial immune systems [Gao and Wang, 2009]. Some of these algorithms may enjoy an increasing popularity in the near future for real world multiobjective optimization problems.

As mentioned earlier, the idea of Pareto ranking has been widely used for extending evolutionary and population-based algorithms to handle multiobjective optimization tasks. In sections 1.6 and 1.8, we will describe how our ant colony optimization and differential evolution algorithms can be used for solving multiobjective history matching problem, based on Pareto ranking.

6.1.3.3.3 Other Methods in A Posteriori Group

We have focused on evolutionary and population-based optimization methods in the a posteriori category for solving multiobjective optimization problems. There are also other methods in this category like the ε-constraint approach [Haimes et al. 1971] where one of the objective functions is selected to be optimized and others are converted into constraints.

We should remember that outside the three categories of prior, progressive and posterior preference approaches, there is a fourth group of methods, so called “no preference”

methods. These types of methods do not use any preference information from the decision maker. Examples of this category are the min-max formulation [Lin, 2001] and the multiobjective proximal bundle (MPB) method [Miettinen, 1999]. These methods will provide a single optimum solution and no attempt is made to find multiple Pareto-optimal solutions. No-preference methods have not been used widely in engineering problems. Figure 6 summarizes different ways of handling multiple objectives.

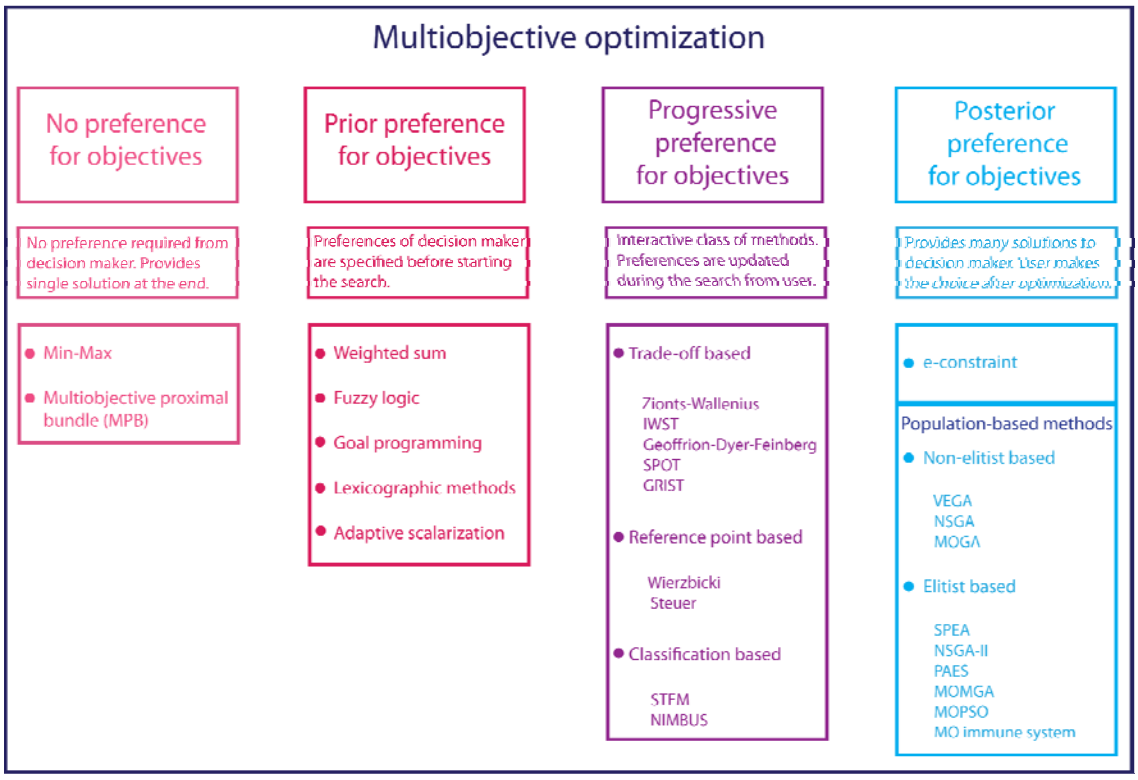

Figure 6: Classification of different methods used in multiobjective optimization

The solution procedure of a multiobjective optimization problem has two phases:

1) Search for the solutions: At this stage, different algorithms are used to explore the parameter and objective function space and find a set of possible solutions. This step is covered in this chapter of thesis using multiobjective ant colony optimization and differential evolution algorithms.
2) Selection of solution: Once we have produced a set of solutions for our multiobjective problem, we will need to make our decision considering different trade-offs and based on high level information. This step involves the process of multi-criteria decision making and it is often necessary to ask a (human)

decision maker to express his/her final preferences. This step is beyond the scope of this chapter.

6.1.4 Applications of Multiobjective Optimization

Most real world problems are multiobjective. We try to minimize the cost of the car we buy, while we aim to maximize our comfort. The same concept applies to other problems we face everyday in real world applications. Beside the original applications in mathematical function optimization and computer science [Coello et al. 2007, Tan et al. 2005], there are numerous applications of multiobjective optimization algorithms for solving engineering challenges. In this section, we shall briefly review these applications, with a focus on geoscience and petroleum engineering domain.

6.1.4.1 Engineering Optimization

Evolutionary multiobjective optimization algorithms have received a great attention in different engineering fields. They have been used to select flight routes to minimize fuel consumption and maximize passenger comfort [Alam et al. 2006]. Azzam proposed a multiobjective ant colony approach for reactive power compensation in electric systems [Azzam and Mousa, 2009]. These algorithms have also been used in civil engineering, for example for the design of space-frame structures [Lagaros et al. 2005] or to control water reservoir operations considering economic, social and environmental limits [Castelletti et al. 2008]. There are also numerous examples of multiobjective evolutionary algorithms in chemical engineering. A nondominated sorting genetic algorithm was used to optimize performance of a steam reformer [Rajesh et al. 2000]. Multiobjective differential evolution has been applied to the optimization of chemical processes [Angira and Babu, 2006]. A good review of multiobjective optimization algorithms for chemical engineering applications can be found in Bhaskar et al. [2000]. General Electric (GE) employs a multiobjective evolutionary algorithm (MOEA) for maximizing the efficiency of coal-fired boilers while minimizing the emissions [Friedemann et al. 2008]. Sadeghzadeh [2009] highlighted the importance of using a multiobjective vision when estimating fluid flow parameters in porous media. He optimized a weighted sum objective function using a gradient based method coupled with a numerical simulator to identify fluid content and pressure head parameters in flow through variably saturated porous media. Yapo presented a multiobjective optimization algorithm based on Shuffled Complex Evolution (SCE-UA) for perturbing

properties of a hydrologic model and obtaining a match between observed and simulated variables [Yapo et al. 1997]. These ideas were followed by other researchers in an attempt to improve the parameter estimation and diversity of the models [Confesor and Whittaker, 2007, Shafii and De Smedt, 2009].

6.1.4.2 Geoscience Applications

Dal Moro and Pipan [2007] used Multiobjective Evolutionary Algorithm (MOEA) for the joint inversion of seismic surface wave dispersion curves and reflection travel times. Boomer and Brazier [2009] also proposed a new approach to obtain velocity models from inversion of seismic data based on a nondominated sorting genetic algorithm (NSGA-II).

6.1.4.3 Petroleum Engineering

Although multiobjective optimization approaches have been widely used in other engineering problems, their application to petroleum engineering problems are still limited. Traditionally the oil industry uses some form of weighted sum approach for handling multiple objectives without considering the limits of this method.

The need for handling many objectives in petroleum engineering was outlined long time ago. Harrison and Tweedie [1981] stated that multiple and often conflicting economical objectives may pose challenges in field developments and proposed an analytic framework to facilitate the decision making process in selecting optimal production policy. Their approach was based on summing up conflicting criteria with predetermined importance (weights).

Rahman et al. [2001] used a multiobjective optimization approach for the design of hydraulic fracturing where he formed a global objective function by summing four objective functions to maximize production and NPV while minimizing treatment cost. The concept of multiobjective optimization was used to balance the energy and force in drilling bits [Perdue, 2002]. The multiobjective genetic algorithm (MOGA) has been applied for the optimization of an ensemble of neural networks and then the tuned ensemble was used to process a pulsed neutron log data [Chen et al. 2004].

Ray and Sarker [2006] used the non dominated sorting genetic algorithm (NSGA-II) to minimize injected gas and maximize produced oil in gas lift operation. Park et al. [2006] presented a multiobjective fuzzy optimization algorithm for the optimal design of pipelines. The goal was to minimize the pipeline investment while maximizing gas delivery. They stated that one of the major benefits of a fuzzy approach is that it can handle uncertainty arising from imprecise description of the information gathered from the field.

Aristeguieta [2008] proposed a decision support system for E&P portfolio optimization based on a multiobjective algorithm. The approach is based on models of risk and return considering different technical and financial objectives. He showed how the Pareto front obtained in this approach helps to understand the complex tradeoffs among different investment portfolios, subject to budgetary constraints. He used multiobjective genetic algorithm with linear constraints (MOGOL) [Medaglia, 2003] for this purpose.

The need for multiobjective optimization has also been highlighted in real field management practices. For example drilling wells in high quality areas of a reservoir may help to reduce the number of producers to get a target production, but also may violate depletion criteria and result in premature water breakthrough, poor sweep efficiency and ultimate recovery [Pham et al. 2008]. Saudi Aramco uses the concept of multiobjective decision making systems (MODM) introduced by Rudduck et al. [2006] to optimize well locations. This approach was based on an analytical well planning model coupled with a response surface and experimental design framework. In this framework, the decision maker (DM) defines a set of possible options and important criteria and then builds a table with decisions and their alternates and ranks them based on their importance weighting. This approach does not use any Pareto dominance or ranking concept and is basically a breakdown of a complex decision making process into smaller and simpler pieces. Saudi Aramco has also implemented a multiobjective optimization procedure for their drilling operations. For this purpose, a goal programming approach is in use to maximize expected production, such that transportation cost is less than a predefined goal [Irgens and Lashar, 2007].

Cardoso addressed the importance of multiobjective optimization in waterflooding projects [Cardoso, 2009]. To minimize water injection and maximize oil production, he

combined these two objectives using a linear weighting approach to make a single objective function. In this approach, as he stated, one should vary the relative weighting of two terms in order to generate the convex portion of the Pareto front. Nghiem [et al. 2009] used Pareto dominance concept to analyze the results of water injection operations. Their work did not involve any adaptive optimization algorithm to optimize the injection location and rate. In this framework, all possible combinations of discrete operating conditions (water rate, injection depth, injection period) are tried and Pareto dominance analysis was performed to select the models that satisfied trapping indexes.

van Essen et al. [2009] proposed a hierarchical production optimization framework which considered long and short term objectives. In this approach, the main objective was to maximize NPV over a long period and in a lower level optimization, the short-term operational performance. They highlighted the difficulty of assigning weights in the balanced (weighted sum) approach. Wang [2003] also used a similar approach for production optimization by considering well operational conditions. He also compared this approach with the objective sum method and concluded that a hierarchical approach is more successful in obtaining solutions on the Pareto front.

In the history matching area, from the first attempts to consider multiple match criteria [Moore and Clark, 1988], the necessity of formulating history matching as a multiobjective problem has been noticed. For a long time, the simple objective sum approach was used to re-formulate multiobjective history matching as a single objective problem. However in recent years this view has started to change and the industry is becoming more aware of the value of the multiobjective optimization algorithms in tackling history matching problems.

An early attempt to apply a multiobjective optimization algorithm to the history matching problem was made by Schulze-Riegert et al. [2007]. They applied the Strength Pareto Evolutionary Algorithm (SPEA) to find Pareto front solutions during history matching. Recently Han et al. [2010] used the nondominated sorting genetic algorithm (NSGA-II) for history matching in a waterflooding project. They used a two dimensional two-phase model with three production and one injection well. Four objective functions were formulated for watercut in production wells and the bottomhole pressure (BHP) of the injection well. History matching was carried out to

900 days and predictions were made to 2161 days. This work did not quantitatively analyze the uncertainty of the predictions. The history matched models with misfit values less than the one obtained by the weighted sum approach were used to assess the prediction capability. Their results showed instability in the water saturation front obtained in the history matched model.

To summarize, multiobjective optimization plays an important role in many practical petroleum engineering applications. From a solution point of view, however, most of the approaches taken to tackle these problems are based on summation of objective functions or considering objectives separately and employing some kind of goal programming approach. The limitations of these approaches have been described in previous sections. Next, we show how current single objective optimization algorithms can be extended to effectively handle many objectives and we will discuss the added-value from Pareto multiobjective optimization.

6.1.5 Multiobjective Optimization Using Ant Colony Optimization

As mentioned earlier, the original ant colony optimization algorithm was proposed for discrete combinatorial optimization and later was adopted for solving continuous problems. Based on this evolution, two classes of multiobjective ACO algorithms can be identified in the literature.

6.1.5.1 Discrete Multiobjective ACO

The first group of works on the extension of ACO for multiobjective problems is in the combinatorial optimization domain. In most early works, optimization criteria were weighted by their relative importance. Mariano and Morales [1999] presented a multi objective ant colony optimization (MOAQ) where for each criterion of the problem there exists one colony of ants. In this algorithm, members of each colony receive a partial solution from members of other colonies and then attempt to further improve this solution with respect to one of the objectives of the problem. Iredi et al. [2001] introduced a bi-objective version of ant system for multiobjective optimization. In this algorithm, m ants are divided into w colonies and these colonies get heterogeneous weighting regarding the objectives of the problem. The weighting parameter dictates the relative importance of the objectives in the problem. Later Ji and Xie [2008] introduced the Multi-Objective Ant Colony System (MOACS) and applied it to the optimal control

of casting operation of steel. In the MOACS algorithm several pheromone matrices were considered for each of objectives. A similar structure has been used in the m-ACO algorithm proposed by Alaya et al. [2007]. Based on the work of Iredi, Hackel et al. [2008] presented a multiobjective ant colony algorithm for optimizing more than two objectives. In their approach the number of ants in each of the sub-colonies was determined according to the number of objectives in the problem and the granularity of the weights between the criteria. Similar algorithms in terms of weighting the objective functions have been proposed in the literature; examples include: Multiple Objective ACO Metaheuristic (MOACOM) [Gravel et al. 2002], ACO Approach to Multiple Objectives (ACOAMO) [McMullen, 2001], the SACO algorithm [T'Kindt et al. 2002] and Multi colony Ant Colony System for Vehicle Rooting Problem (MACS-VRP) [Gambardella et al. 1999].

Guntsch [2004] introduced the population based ant colony optimization (PACO) and modified it for multiobjective discrete optimization problems. In this algorithm, a set of non-dominated solutions were found and stored as the "super population". This super population was used to construct the pheromone information for the ants. The probability of selecting a solution in the discrete space was determined by aggregating the individual probabilities regarding each criterion. This form of probability update reflects to what extent the search for a new solution in the non-dominated front should go in the direction of the implied criterion. Cheong and Tan [2008] presented the Multi-Objective Multi-Colony Ant Algorithm (MOMCAA). This algorithm uses an island model with heterogeneous colonies. During the optimization, each group of ants construct a solution in the discrete solution space. The proposed solutions are stored in an external archive. After appending the current population, first the repeated solutions are deleted and then a Pareto ranking is performed on remaining members. Low rank solutions (dominated members) are removed to keep the size of archive fixed. Angus [2009] presented a taxonomy of the multiobjective ant colony optimization algorithms. He classified these algorithms based on their pheromone matrix structure, solution construction and evaluation, update and decay of pheromone and Pareto archival mechanisms.

To summarize, discrete multiobjective ant colony optimization algorithms use the original concept of graphical representation of the problem and making decisions based

on pheromone values on the nodes of graph. These methods differ in the number of colonies assigned for the objectives, the pheromone update rule and the probabilistic selection criteria in each decision node.

6.1.5.2 Continuous Multiobjective Ant Colony Optimization

Early attempts to apply ant colony optimization for continuous multiobjective problems used objective function aggregation. Baskar et al. [2004] used a modified CACO algorithm [Bilchev and Parmee, 1995] to optimize a global objective function in a surface grinding operation. Another CACO-inspired algorithm for multiobjective optimization was proposed by Prakash et al. [2002]. Socha [2008] argued that the proposed algorithm by Bilchev and Parmee [1995] does not follow the original formulation of the ACO. The nest concept was introduced in CACO, which does not exist in the original ACO. Also, CACO does not perform an incremental construction of solutions. Based on the above discussions, multiobjective ant colony optimization algorithms based on CACO may not be considered as extensions to the original ACO algorithm.

One of the most successful extensions of the ant colony framework for continuous optimization is the ACO_R algorithm [Socha and Dorigo, 2008]. We have previously discussed the principles of the ACO_R algorithm and applied it to single-objective history matching. There has been little attempt in the literature to extend ACO_R algorithm to multiobjective problems. The only published extension which directly applies this algorithm to multiobjective optimization problems was by Khalidji et al. [2009]. The authors proposed a dynamically weighted continuous ant colony optimization (DW-ACO_R) for bi-objective problems based on the original ACO_R algorithm. The idea behind this algorithm was a dynamic weighting of objective functions for aggregation using Tchebychev norm. The weights were generated in each iteration according to the following rule.

$$\left.\begin{aligned} w_1(t) &= \sin\left(\frac{2\pi t}{T}\right) \\ w_2(t) &= 1 - w_1(t) \end{aligned}\right\} \qquad (4)$$

where *T* denotes the period of weight changes. By changing the weights, various points on the Pareto front can be obtained.

Population-based ACO for Multi-objective Function Optimization (PACO-MOFO) was introduced by Angus [2007]. This work was based on the population-based ant colony optimization (PACO) algorithm presented by Guntsch [2004] and also borrows some ideas from the ACO_R algorithm [Socha and Dorigo, 2008]. In this algorithm, first the population of the ants are initialized randomly (size = number of ants). Then a Pareto ranking of the population is performed. Fitness of each population member is set to the inverse of its rank and then the fitness value is adjusted according to fitness sharing concept. Then using a roulette wheel selection with replacement, a solution is probabilistically selected among the population. Then as in the ACO_R algorithm, the Gaussian probability function is computed and a new solution is generated. Unlike the ACO_R algorithm which adaptively calculates the standard deviation of the Gaussian distribution, in PACO-MOFO, this value is calculated based on a convergence factor during optimization. This convergence factor is calculated based on the following equation:

$$\text{Convergence factor} = (\sin(\pi / 2 \times \text{remaining evaluations} / \text{maximum evaluations}))^2 \quad (5)$$

Then a subset of solutions is selected from the main population. If the new candidate dominates the closest matching solution in the selected subset population, the candidate solution replaces it, otherwise the solution is discarded. The crowding replacement in this algorithm helps to have diverse solutions from the objective space and the fitness sharing promotes an even sampling of the objective space.

In the following section, a novel extension of ACO_R for multiobjective optimization problems based on Pareto ranking of solutions will be described.

6.1.6 Pareto Ranking ACO_R ($PRACO_R$)

$PRACO_R$ is based on the ACO_R algorithm proposed by Socha and Dorigo [2008]. Generation of new solutions based on a normal distribution and the computational method used to generate the elements of this distribution are the same as the original ACO_R algorithm. $PRACO_R$ is different in the way that the pool of candidate solutions

are formed and accessed in order to generate new solutions. In this study, we use $PRACO_R$ algorithm to handle two objectives.

$PRACO_R$ starts the optimization by generating k random solutions to initially fill the solution archive (size k). These solutions are evaluated against the objectives of the problem and fitness values are computed. We should remember that the original ACO_R algorithm sorts the current solution archive according to objective function values and works based on ranks of the population members. The better the objective function value, the higher the rank of the solution. For generating the next set of solutions, we need to rank the solutions in the $PRACO_R$ archive. Here, we use a Pareto ranking mechanism (MOGA style) to count the number of solutions that dominate each member of the population. We consider two different strategies for continuing the optimization cycle: $PRACO_R$-1 and $PRACO_R$-2.

In the $PRACO_R$-1 algorithm, we multiply the Pareto rank of each member by the sum of the objective function values. For each individual in the solution archive of the $PRACO_R$-1, we have:

New objective function value = Pareto rank × (sum of objective function values)

This update of the objective function values according to their Pareto ranks is the only difference between $PRACO_R$-1 and the original ACO_R algorithms. The new objective function values are used as the fitness scores in the $PRACO_R$-1 algorithm. We sort the solution archive based on the updated objective function values. The current solution archive is kept in the memory for future reference. Then, following the rules of the original ACO_R algorithm, we generate m new members and add them to current solution archive. This forms a solution archive with size (k+m). Next, we perform a Pareto ranking on this archive and update the fitness values. In order to reduce the solution archive size to its original value (k), we remove the m worst solutions with highest updated objective function values. The new solution archive (size k) is used to generate the next set of solutions. This procedure is repeated until the termination condition (maximum number of function evaluations) is met.

To summarize the procedure of $PRACO_R$-1 algorithm, we have:

1. Initialize the population with *k* random solutions and obtain objective function values
2. Perform a Pareto ranking and update the fitness values by multiplying the Pareto rank to sum of the objective function values

Repeat steps 3-8 until the maximum number of function evaluations is reached

3. Generate *m* members and evaluate objective functions. Add these solutions to the archive (size *k*+*m*)
4. Use the Pareto ranking (MOGA style) to assign dominance scores to the members of the solution archive
5. Multiply the Pareto ranks to the sum of the objective function values and update the fitness values in solution archive
6. Sort the solution archive (*k*+*m*) based on updated objective function values
7. Remove the *m* worst solutions to keep the solution archive size fixed.
8. Use updated solution archive to generate new solutions using the original ACO_R algorithm

In $PRACO_R$-2, instead of multiplying Pareto ranks by the objective function values, we directly use Pareto ranks as the fitness values. After evaluating members of the initial solution archive for both objectives, we perform a Pareto ranking on this solution archive and sort it based on the Pareto ranks. This places nondominated solutions at the top of the solution archive. Then *m* new solutions are generated using normal ACO_R operations and we add them to the current archive. This increases the size of the archive to (*k*+*m*). Then the second Pareto ranking is applied to this archive and members are sorted according to their Pareto ranks. Next, we remove the worst *m* solutions from the bottom of the solution archive in order to reduce the archive size to its original (k). Then we use the individuals in this updated archive to produce the next set of solutions. This is repeated until the termination condition (maximum number of function evaluations) is met.

To summarize the procedure of $PRACO_R$-2 algorithm:

1. Initialize the solution archive with *k* random members and evaluate the solutions against the objectives of the problem
2. Perform a Pareto ranking for solutions based on the Pareto dominance concept

Repeat steps 3-7 until the termination condition (maximum number of function evaluations) is met

3. Generate *m* solutions, evaluate them against the objectives of the problem and form a solution archive of size (*k*+*m*)
4. Use the Pareto ranking mechanism (MOGA style) to assign dominance scores to members in the solution archive
5. Sort the archive (size = *k*+*m*) according to Pareto ranks
6. Keep only the first *k* solutions (delete the *m* worst solutions with highest dominance scores)
7. Use this population to generate the next set of solutions using normal ACO_R operators

In both strategies of the $PRACO_R$ algorithm, we only use the calculated fitness values based on Pareto ranks ($PRACO_R$-1) or the Pareto ranks only ($PRACO_R$ -2) and we do not employ any niching mechanism.

6.1.7 Multiobjective Optimization Using Differential Evolution

Differential evolution has been extensively studied in the multiobjective optimization domain. Since the original DE algorithm was designed for continuous parameter optimization, unlike ACO, it has been the focus of more research to extend it to continuous multiobjective optimization. Mezura-Montes et al. [2008] classified different DE algorithms for solving multiobjective problems into three main groups:

1. Non-Pareto methods
2. Pareto-based approaches
    a. Using Pareto dominance
    b. Using Pareto scoring
3. Combined methods

6.1.7.1 Non-Pareto Methods

Non-Pareto-based algorithms use the classical methods to handle multiple objectives like combination of objective functions, etc. In all of the algorithms proposed in this category no change is made to the working mechanism of differential evolution. As an example we can reference Babu and Jehan [2003] where they used weighting factors for bi-objective optimization problems. They also used another approach where the second objective function was considered as a constraint using a penalty function. Another example in this group is the MODE/D algorithm proposed by Li and Zhang [2006]. In the Multiobjective differential evolution based on decomposition (MODE/D), the weighted Tchebycheff approach is used to decompose the multiobjective problem into multiple sub-problems. These decomposed sub-problems are minimized in a single run.

6.1.7.2 Pareto-Based Methods

Pareto-based approaches benefit from an internal mechanism that rewards the solutions based on Pareto optimality concept. There are two subgroups in this category: Pareto dominance and Pareto scoring approaches.

6.1.7.2.1 Pareto Dominance

Shortly after the introduction of the original DE in 1995, Chang et al [1999] proposed a multiobjective differential evolution algorithm for tuning fuzzy membership functions. They used an external archive to store non-dominated solutions. Each new candidate solution was compared with the solutions in the Pareto-optimal set. If the candidate solution was dominated by any of the Pareto-optimal solutions, comparison with the rest of optimal set was stopped and the candidate was discarded. If the candidate dominated at least one solution in the set, then the dominated solution was removed from the set and the candidate solution replaced it. This algorithm also used a fitness sharing mechanism to maintain diversity of the obtained solutions.

The Pareto Differential Evolution (PDE) algorithm was proposed by Abbass and Sarker [2002]. PDE employs differential evolution operators to create new individuals and then keeps only the non-dominated solutions as the pool for next generation. In PDE, the starting population is initialized according to a Gaussian distribution. After initialization, only non-dominated solutions are kept. Then three members are randomly chosen and one new solution is generated. This solution only replaces the parent if

dominates it. This process is repeated until the population is completed. If the size of the nondominated solutions exceeds a predefined threshold, a distance metric is applied to discard the parents which are located in a close neighbourhood of each other. Abbass [2002] also proposed the self adopted version of PDE algorithm (SPDE) in which the algorithm tunes its crossover and mutation operator. Another work by Abbass is the Memetic Pareto Artificial Neural Networks (MPANN) which is the enhanced version of PDE algorithm with Back-Propagation (BP) local search algorithm to speed up the convergence rate [Abbass 2001].

Generalized Differential Evolution (GDE) was proposed by Lampinen [2001]. In the GDE algorithm, the trial vector enters the population if it dominates the parent vector. There is also an improved version of GDE where a crowding distance measure is used to prevent the population from being concentrated in a specific region [Kukkonen and Lampinen, 2004].

Huang et al. [2005] introduced a multiobjective differential evolution with an external archive mechanism to store the promising solutions based on Pareto dominance concept. This archive had a predefined limit in order to restrict it from growing indefinitely. Harmonic average distance was also employed in this algorithm to measure crowding distance and guide the search towards less populated areas.

Babu et al. [2005] presented a differential evolution algorithm based on Pareto dominance for multiobjective optimization. In the MODE algorithm, in each generation, dominated solutions were removed from the population and only non-dominated members were allowed to undergo DE operations. In this algorithm, the scaling factor (F) was randomly selected between 0 and 1. Although this approach provided good results, it suffered from a major drawback. The domination condition was checked before the candidate solutions were passed to the next generation. This will gradually reduce the size of the population during optimization. To overcome this issue, modified versions of MODE algorithm was introduced by Babu and colleagues. In the MODE-2 algorithm, after removing dominated solutions in each generation, the same number of random solutions were added to the current population to maintain a population size constant. In MODE-3 algorithm, the population was initialized randomly. Then the original DE operations were performed until the stopping criterion was met or Pareto

front was obtained. After the last generation, non-dominated sorting was performed to remove the dominated solutions [Babu et al. 2007]. Babu and Gujarathi [2007] also proposed another version of the MODE algorithm called Elitist-Multiobjective Differential Evolution (E-MODE) which benefited from a crowding distance mechanism to increase diversity of the solutions.

Zhang and Sanderson [2009] proposed an adaptive version of multiobjective differential evolution, called JADE. In the selection process of JADE, candidate solutions are compared according to Pareto dominance and crowding density estimation at a lexicographic order. A solution is considered better if it dominates the other solution or if it has a lower crowding density when dominance comparison ties.

6.1.7.2.2 Pareto Scoring

One of the most popular ways for selecting the best individuals for the next generation is non-dominated scoring which can be based on Pareto ranking or sorting. Madavan [2002] introduced one of the first extensions of differential evolution for multiobjective problems based on Pareto sorting mechanism. The Pareto-Based-Differential Evolution (PBDE) modifies the basic DE algorithm by incorporating the fast nondominated sorting scheme introduced by Deb et al. [2000]. In PBDE, once the new population is generated, it is mixed with the parent population and a nondominated sorting is performed on the combined population (size = 2*N*). It calculates the nondominated score using Pareto-based sorting and diversity rank according to the crowding distance metric for each of the combined population members. Madavan studied two variants of the proposed algorithm. In the first version, there is a one-to-one comparison between each pair of parent and off-spring solutions. Each off-spring solution is compared with its parent and only proceeds to the next generation if it has higher nondominated rank or, the same or higher diversity rank. He reports that this variant is not efficient because not much elitism is incorporated into the working mechanism of the algorithm. He then proposes the second variant where simply takes the best individuals according to the non-domination and diversity scores. This version was proved to be very efficient and produced favorable results.

Xue et al. [2003] introduced the Multi-Objective Differential Evolution (MODE) algorithm. They proposed a variant which is similar to the *best/1/bin* scheme of the

original algorithm. In each generation a Pareto sorting was performed according to the Goldberg approach [1989]. If the trial vector was nondominated, it was selected as the best member to build the next population. If this solution was dominated, a set of nondominated members can be identified and the "best" was simply selected from the set of non-dominated solutions. In this algorithm the fitness of each solution was calculated based on Pareto-based ranking and then reduced with respect to the crowding distance value of that solution. Crowding distance penalized the fitness of individuals to improve the selection process. Nondominated Sorting Differential Evolution (NSDE) was proposed by Iorio and Li [2004]. This technique was built based on NSGA-II algorithm. In NSDE algorithm, N off-springs were generated from the selected parents. After objective function evaluation, the new solutions were combined with the parent generation. The combined population was sorted according to dominance ranks and crowding distance. The population size was reduced to N, after deleting the worst solutions, and *current-to-rand/1* strategy was used to generate new candidates.

Robic and Filipic [2005] introduced Differential Evolution for Multi-Objective Optimization (DEMO) algorithm. In DEMO, each new solution was compared with its parent and if the candidate dominated the parent, it will replace the parent. If the candidate and the parent were nondominated regarding each other, the new solution was added to the population. The above loop was repeated until N solutions were generated. The total population size was between N and $2N$. To reduce the population size, a nondominated sorting with crowding distance metric was performed. In this work, they also proposed two other versions of DEMO algorithm, where the child was compared with most similar individuals (in parameter space or objective space), instead of comparison with direct parent. Later Zamuda [et al. 2007] proposed a version of DEMO where algorithm tuning parameters, F and C_r, were adaptively adjusted to appropriate values.

Peng [et al. 2008] proposed Opposition-Based Multiobjective Differential Evolution (OMODE). The OMODE algorithm was based on the concept of opposition-based learning (OBL) which was developed by Tizhoosh [2005]. The main idea behind OBL was the simultaneous consideration of an estimate and its corresponding opposite estimate in order to achieve a better approximation of the current candidate solution.

Gong [et al. 2009] introduced a multiobjective differential evolution where the initial population was selected using orthogonal design method with quantization techniques. The aim was to sample a small but representative set of combination of parameters in the initial stage which scatter uniformly in the space of all possible parameter combinations.

6.1.7.3 Combined Methods

Combined methods use different schemes like global and local search, Pareto dominance and ranking to deal with multiobjective optimization problems. Parsopoulus et al. [2004] introduced the Vector Evaluated Differential Evolution (VEDE) approach. In this method the population was divided into M sub-populations and each of these smaller populations were evaluated for one of the objectives in the problem. These sub-populations were connected in a ring topology and exchange the information after objective function evaluation to select the next generation. VEDE used a Pareto selection concept to favor survival of the non-dominated individuals. Due to information exchange mechanism, only DE schemes that use the best member can fully benefit from VEDE. Xue et al. [2005] extended their previous work on multiobjective differential evolution (MODE) by introducing a fuzzy logic controller (FLC) to the algorithm. Within this framework, FLC was used to tune the parameters of algorithm, including scaling factor and crossover value. Chen et al. [2008] proposed a generalized multiobjective DE (GDE) combined with estimation of distribution algorithm (EDA). This approach simultaneously used global information of the population extracted by both EDA and DE. EDA extracts the global statistical information about the population and this information helps the GDE to guide its search and avoid trapping in local minima. Zamuda et al. [2009] also presented a multiobjective differential evolution with adaptive parameters and a local search scheme (DECMOSA-SQP). This algorithm used sequential quadratic programming (SQP) as the local search method.

6.1.8 Differential Evolution for Multiobjective Optimization Based on Pareto Ranking (DEMOPR)

In this section we propose an extension of the differential evolution algorithm for multiobjective optimization based on a Pareto ranking mechanism (MOGA style). This approach also borrows ideas from the Pareto-based differential evolution (PBDE) algorithm [Madavan, 2002]. According to the conclusions of Mezura-Montes et al.

[2008] most of the multiobjective differential algorithms use the popular *DE/rand/1/bin* scheme in their search. Further to *DE/Rand* strategy, we also use *DE/Best* scheme in our proposed method.

The DEMOPR algorithm, like original differential evolution, is initialized using N random solutions. We continue the normal operations of DE and build the next generation of solutions. Then a pool of combined population of initial random solutions and first generation is formed (size $2N$). These solutions are ranked based on the Pareto dominance concept. The Pareto ranking procedure is like the MOGA algorithm where each individual receives a rank which shows the number of population members that dominate the selected solution. This is one of the differences between DEMOPR and PBDE algorithm in which nondominated sorting (NSGA style) is applied to identify different layers (fronts) of dominance. At the end of Pareto sorting in DEMOPR, all members of the population (size $2N$) will receive a dominance rank. Ranking on the $2N$ population will obviously require more computation time than the approach that just ranks the off-spring population, but will allow a global dominance check for both parent and off-spring solutions. Also this additional time is negligible in comparison with the time required for flow simulations in history matching. From this point onwards, we take two different strategies, DEMOPR-1 and DEMOPR-2. These strategies are different in the form of using obtained Pareto ranks in assigning fitness values to members of the population.

The first strategy, called DEMOPR-1, is a quick-fix approach where the Pareto rank of each population member is multiplied to the sum of objective function values. For each individual in the population, we can express this as:

New objective function value = Pareto rank × (sum of objective function values)

This new objective function is used as the fitness value in the DEMOPR-1 algorithm. For example, for a solution which is not dominated by any member of the population (Pareto rank =1), the new fitness value would be same as the sum of objective function values and for a solution which is dominated just by one individual, the updated fitness value is two times larger than the sum of objective functions. Next, the population (size $2N$) is sorted according to the updated fitness values. In order to reduce the population

size to its original (N), the first N solutions in the combined pool are kept and the rest of the population is discarded. The new population (size N) is passed to differential evolution algorithm to select the vectors of the next population. The original procedures of differential evolution (vector selection, applying scaling factor, mutation) are performed on this updated population. We have extended two strategies of the DE algorithm to multiobjective optimization within the DEMOPR framework. In the *DEMOPR/Rand* strategy, the vector of solutions are randomly selected from the new population. In the *DEMOPR/Best* strategy, the best solution of the population is selected as the base vector. After generating the new set of solutions, the population is evaluated against the objectives of the problem. Then the same steps are repeated, the current population is added to previous one, Pareto ranking is performed and the first N best solutions with lower objective function values are kept. This is repeated until the termination condition (maximum number of function evaluations) is met.

To summarize procedures of the DEMOPR-1 algorithm, we have:

1. Initialize the population with N random solutions and obtain the objective function values
2. Perform Pareto ranking and update the fitness values by multiplying the Pareto rank to sum of objective function values

Repeat steps 3-8 until the termination condition (maximum number of function evaluations) is met

3. Combine the current population with previous population and form a pool of $2N$ solutions
4. Use Pareto ranking mechanism (MOGA style) to assign dominance scores
5. Multiply the Pareto ranks to sum of the objective function values and update the fitness scores
6. Sort the $2N$ population based on updated objective function values
7. Only keep the first N solutions
8. Use this population to generate the next set of solutions using normal DE operators

The second approach, which is called DEMOPR-2, directly uses the Pareto ranks instead of the objective function values. After forming the joint population pool and performing Pareto ranking, the population (size 2*N*) is sorted according to Pareto ranks. This sorting places the better solutions (individuals not dominated by many members of the population) in the top of the archive. At this stage to reduce the population size to its original (*N*), the first *N* solutions (dominated by less members in the population) are kept and the rest of the population is deleted. The new population (size *N*) is now used for generating the next set of solutions. Normal DE operations are performed to generate vectors of solutions. DEMOPR-2 also works with *Rand* and *Best* strategies for multiobjective optimization. In *DEMOPR/Rand* strategy, vector of solutions are randomly selected among updated population members to form the next generation. In *DEMOPR/Best* strategy, one of the solutions with Pareto rank of 1 (not dominated by any member of the population) is randomly selected as the base vector. After generating the new set of solutions, the population is evaluated for objectives of the problem. Then the same steps are repeated, current population is added to previous one, Pareto ranking is performed and the first *N* best ranks are saved for producing the next generation of solutions. This is repeated until the stopping condition (maximum number of function evaluations) is met.

To summarize procedures of the DEMOPR-2 algorithm, we have following steps:

1. Initialize the population with *N* random members and evaluate the solutions for objectives
2. Do Pareto ranking (MOGA style)

Repeat steps 3-7 until the termination condition (maximum number of function evaluations) is met

3. Combine current population with the previous one and form a pool of 2*N* solutions
4. Use Pareto ranking mechanism to assign dominance scores for combined population
5. Sort the 2*N* population based on Pareto ranks
6. Only keep the first *N* solutions

7. Use this population to generate the next set of solutions using normal DE operators

We should note two main points in DEMOPR algorithm. First, unlike PBDE and most of the multiobjective DE algorithms, DEMOPR does not use any niching mechanism and only relies on the calculated fitness values based on Pareto ranks (DEMOPR-1) or the Pareto ranks only (DEMOPR-2). Second, for this study, we have developed a bi-objective version of DEMOPR. However handling more than two objectives can be easily added to current framework of the proposed algorithm.

6.2 Test of Developed Algorithms

As previously mentioned, a multiobjective optimization algorithm should have two main characteristics; first converging to PF_{True} and second maintaining diversity of the solutions on the Pareto front. Based on the problem and its constraints, PF_{True} may be connected or disconnected, convex or concave. Each test-bed developed for evaluating performance of multiobjective optimization techniques should consider the behavior of the algorithm in achieving the above mentioned goals.

6.2.1 Zitzler-Deb-Thiele (ZDT) Benchmark

Zitzler et al. [2000] proposed a test suite for multiobjective optimization problems which turned out to be a standard benchmark case among researchers. The test suite contains six functions in the general form of

$$\left.\begin{aligned} &Minimize: \ F(x) = (f_1, f_2), \\ &subject\ to: \ f_2(x) = g(x_2, ..., x_m)h(f_1(x_1), g(x_2, ..., x_m)), \\ &\quad where: x = (x_1, ..., x_M) \end{aligned}\right\} \qquad (6)$$

In this chapter, we have selected first three functions of the ZDT benchmark suite to test our new algorithms. We have selected these three functions because each of them offers a specific type of Pareto front. The effect of convexity and non-convexity of the Pareto-optimal front are studied in the ZDT1 and ZDT2 functions and the effect of discontinuities and disconnectedness in the Pareto-optimal front are examined in the ZDT3 test. Also each of these functions has 30 unknowns. The large number of decision

variables makes ZDT test suite a difficult benchmark problem for optimization algorithms.

6.2.1.1 ZDT1

This test function has a convex Pareto-optimal front and can be expressed as:

$$\left.\begin{aligned} f_1(x) &= x_1 \\ f_2(x,g) &= g(x).(1-\sqrt{f_1 / g(x)}) \\ g(x) &= 1+\frac{9}{n-1}.\sum_{i=2}^{n} x_i \end{aligned}\right\} \quad (7)$$

where n = 30, and $x_i \in [0,1]$. The PF_{True} is formed with $g(x)$ = 1. Figure 7 shows PF_{True} for ZDT1 function (a) and 25000 randomly generated solutions (b).

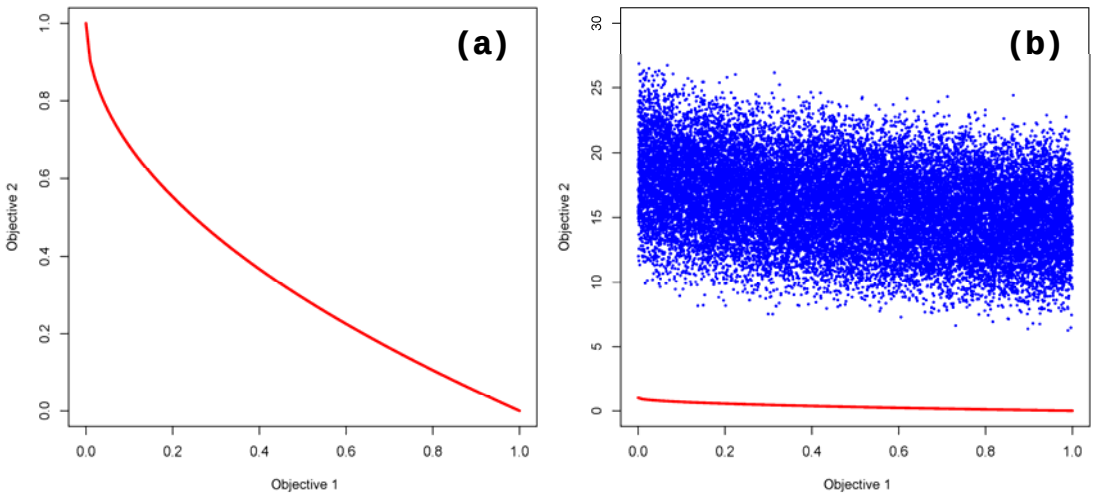

Figure 7: The true Pareto front for ZDT1 test function (a) and 25000 randomly generated solutions for this problem (b)

6.2.1.2 ZDT2

ZDT2 has a nonconvex Pareto-optimal front. This function is in the following form:

$$\left.\begin{aligned} f_1(x) &= x_1 \\ f_2(x,g) &= g(x).(1-(f_1 / g(x))^2) \\ g(x) &= 1+\frac{9}{n-1}.\sum_{i=2}^{n} x_i \end{aligned}\right\} \quad (8)$$

where $n = 30$, and x_i [0,1]. The PF_{True} is formed with $g(x) = 1$. Figure 8 shows PF_{True} for ZDT2 function (a) and 25000 randomly generated solutions (b).

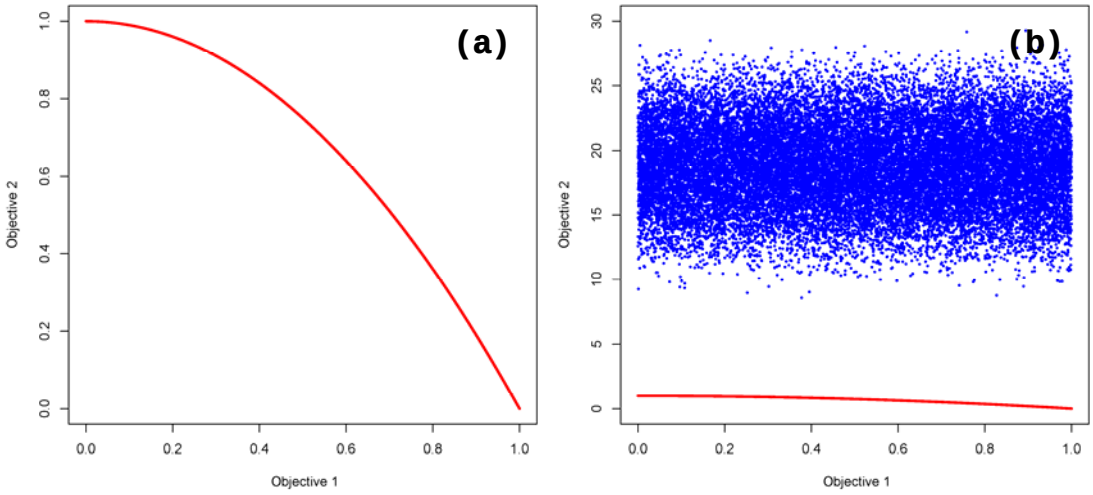

Figure 8: The true Pareto front for ZDT2 test function (a) and 25000 randomly generated solutions for this problem (b)

6.2.1.3 ZDT3

This test function has a disconnected Pareto-optimal front with several convex parts. We can show ZDT3 function as:

$$\left.\begin{aligned} f_1(x) &= x_1 \\ f_2(x,g) &= g(x).\left(1-\sqrt{\frac{f_1}{g(x)}}-\frac{f_1}{g(x)}.\sin(10\pi f_1)\right) \\ g(x) &= 1+\frac{9}{n-1}.\sum_{i=2}^{n} x_i \end{aligned}\right\} \quad (9)$$

where $n = 30$, and x_i [0,1]. The PF_{True} is formed with $g(x) = 1$. Figure 9 shows PF_{True} for ZDT3 test function. The reason of discontinuity in PF_{True} is the *sin*() function. However, parameter space in ZDT3 function is continuous. Figure 9 presents PF_{True} for ZDT3 test function (a) and 25000 randomly generated solutions (b).

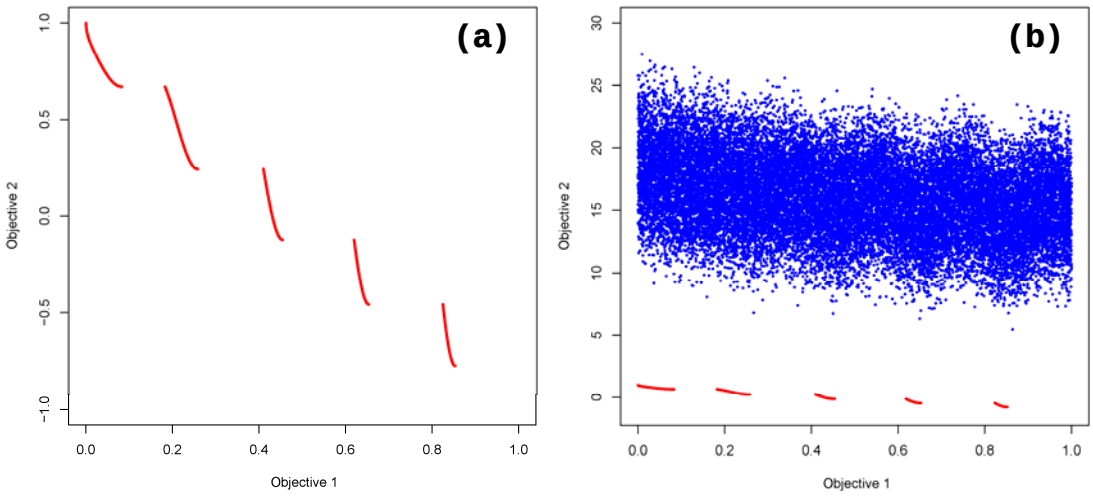

Figure 9: The true Pareto front for ZDT3 test function (a) and 25000 randomly generated solutions for this problem (b)

It should be noted that in all ZDT functions, randomly generated solutions cannot capture the Pareto fronts (i.e. hardly getting numbers around 5 for objective function 2). These examples clearly demonstrate the weakness of random search methods for tackling multiobjective optimization problems.

6.2.2 Results of Algorithms for ZDT Test Functions

In this section, we present the results of applying PRACO$_R$ and DEMOPR algorithms to the ZDT1, ZDT2 and ZDT3 test functions. For each test function, three scenarios are discussed. In the first scenario, we simply sum two objectives of the test functions and optimize the resulted global objective function using the original ACO$_R$ and DE algorithms. The second scenario is the quick-fix approach where for each test function, we multiply the Pareto ranks by the sum of the objective functions (PRACO$_R$-1 and DEMOPR-1). The third scenario involves using the Pareto ranks as the fitness values (PRACO$_R$-2 and DEMOPR-2). The maximum number of function evaluations for all tests was 25000.

6.2.2.1 ZDT1

Table 1 and 2 shows the tuning parameters used for DEMOPR and PRACO$_R$ algorithms for optimization of ZDT1 benchmark function.

Table 1: Tuning parameters of DEMOPR for random and best strategies in ZDT1 test

Strategy	N_p	F	C_r	Generations
Rand	100	0.3	0.7	250
Best	100	0.9	0.5	250

Table 2: Tuning parameters of $PRACO_R$ in ZDT1 test

Number of ants	K	q	ξ	Generations
100	100	0.5	0.5	250

Figure 10 shows the results of several multiobjective optimization algorithms extracted from Zitzler et al. [2000], Xue et al. [2003] and Chowdhury et al. [2009] for the ZDT1 test functions. Zitzler et al. [2000] compares the following algorithms: a random search algorithm (RAND), the Fonseca and Fleming [1993] multiobjective genetic algorithm (FFGA) or (MOGA), Niched Pareto Genetic Algorithm (NPGA) [Horn et al. 1994], the Hajela and Lin [1992] weighted-sum approach (HLGA), vector evaluated genetic algorithm (VEGA), the Nondominated Sorting Genetic Algorithm (NSGA), a single-objective evolutionary algorithm using weighted-sum aggregation (SOEA) and the Strength Pareto Evolutionary Algorithm (SPEA). Xue et al. [2003] compared the multiobjective differential evolution (MODE) with SPEA and Chowdhury et al. [2009] presented the results of a modified predator prey algorithm. All of these algorithms use a population of 100 in each generation and continue the optimization for 25000 function evaluations.

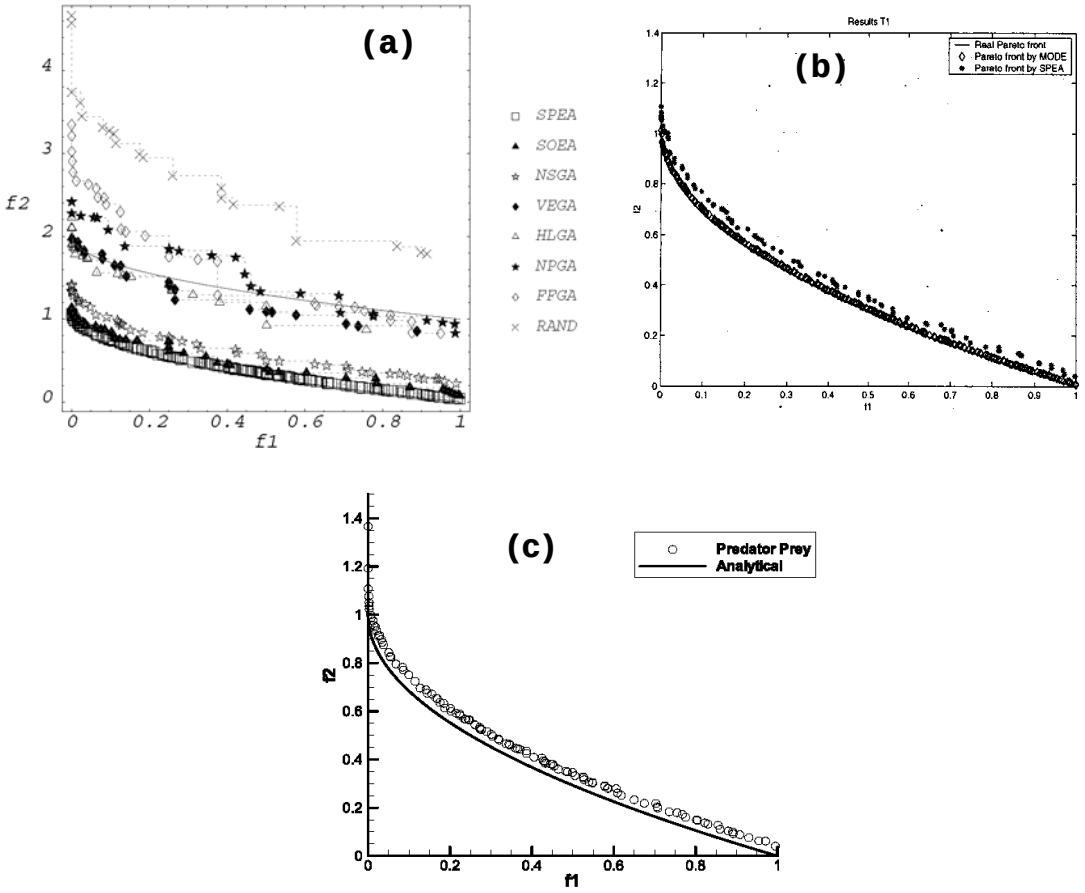

Figure 10: Comparison of different multiobjective algorithms for ZDT1 test function. (a) after Zitzler et al. [2000], (b) after Xue et al. [2003] and (c) after Chowdhury et al. [2009]

Figure 11 presents the results of the ant colony optimization algorithm for the ZDT1 test function. As we can see in figure 11 (a), using the objective sum approach, we can only identify a small part of the whole Pareto front. This figure shows one of the major drawbacks of objective sum approach. Section (b) shows the results of the $PRACO_R$-1 algorithm where the sum of the objective functions is multiplied by the Pareto ranks of the solutions. We see an improvement of performance in approaching the Pareto front by this algorithm. Finally, section (c) of the figure shows the $PRACO_R$-2 approach where we only rely on Pareto ranks. We can see that, in terms of the solution diversity on the Pareto front, the $PRACO_R$-2 algorithm covers the whole front. However, there is a gap between nondominated solutions found by the algorithm and the true Pareto front shown with the red line in the figure.

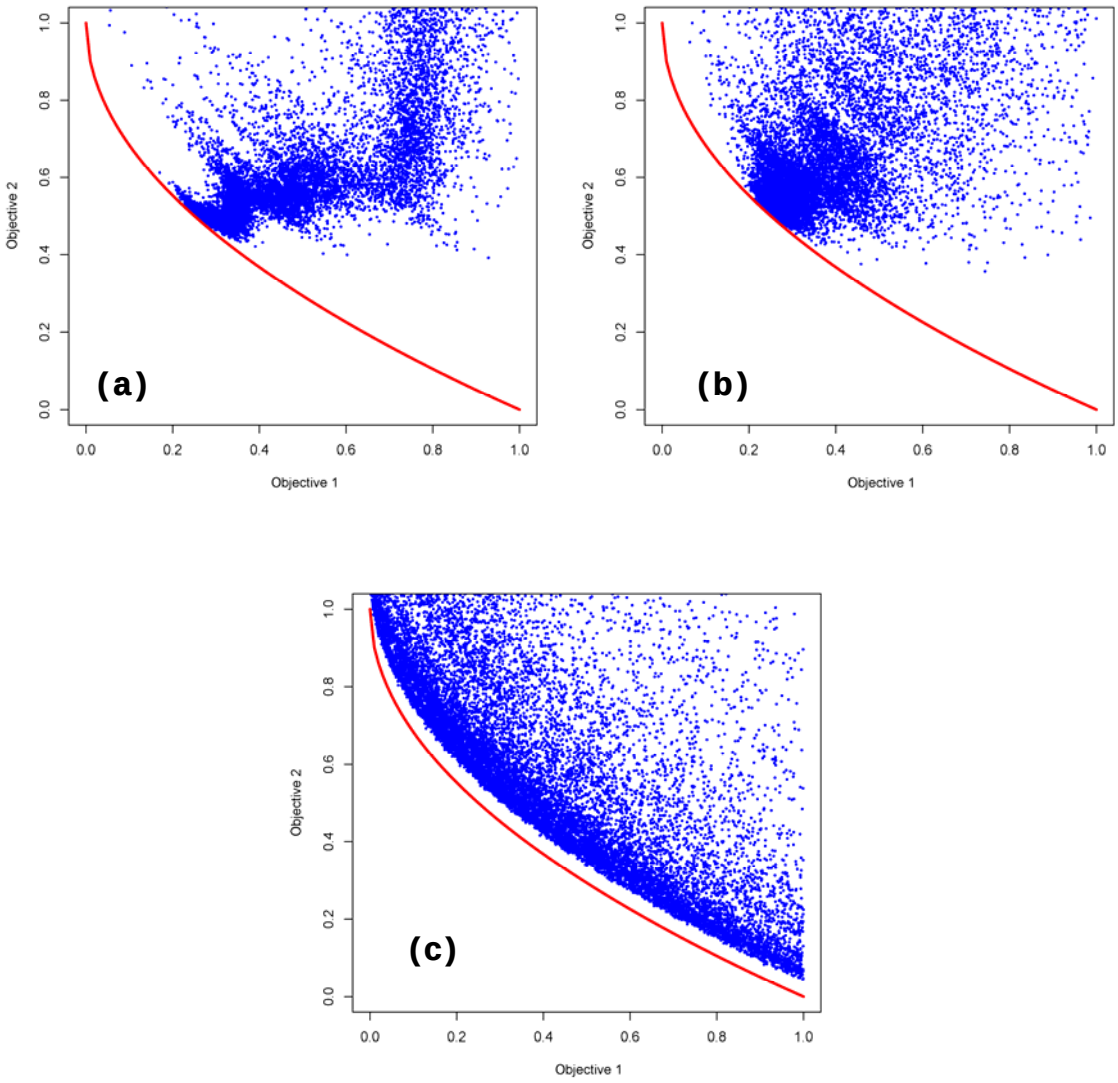

Figure 11: ACO_R results for optimization of the ZDT1 function: (a) using objective sum approach, (b) $PRACO_R$-1 using Pareto rank × objective sum, and (c) $PRACO_R$-2 using Pareto ranks only

Figure 12 shows the DEMOPR algorithm with random strategy for the ZDT1 function. As expected, here the objective sum approach also fails to capture the whole Pareto front. DEMOPR-1 with random strategy is able to capture more regions of the Pareto front and DEMOPR-2 shows the best performance both in getting close to Pareto front and obtaining diverse solutions to cover the whole front.

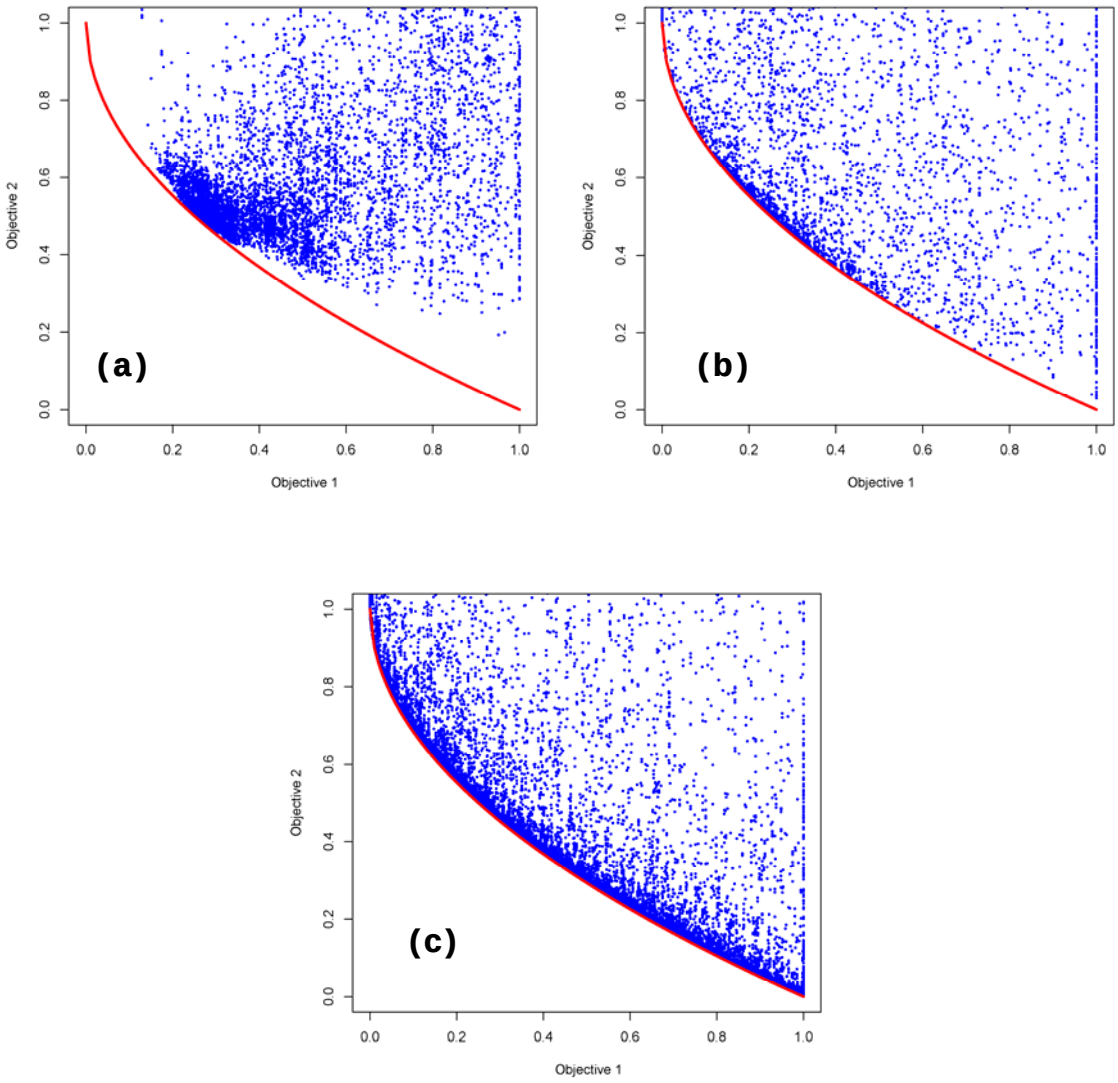

Figure 12: DE-Rand results for optimization of the ZDT1 function: (a) using objective sum approach, (b) DEMOPR-Rand-1 using Pareto rank × objective sum, and (c) DEMOPR-Rand-2 using Pareto ranks only

In figure 13 we present the performance of the DEMOPR-Best algorithm for optimization of the ZDT1 test function. The results of this algorithm are also consistent with DEMOPR-Rand and PRACO$_R$ algorithms. In DEMOPR-2-Best, we have the best coverage of the Pareto front. For DEMOPR-1-Best and DEMOPR-2-Best, red lines showing true Pareto front are removed for better visualization.

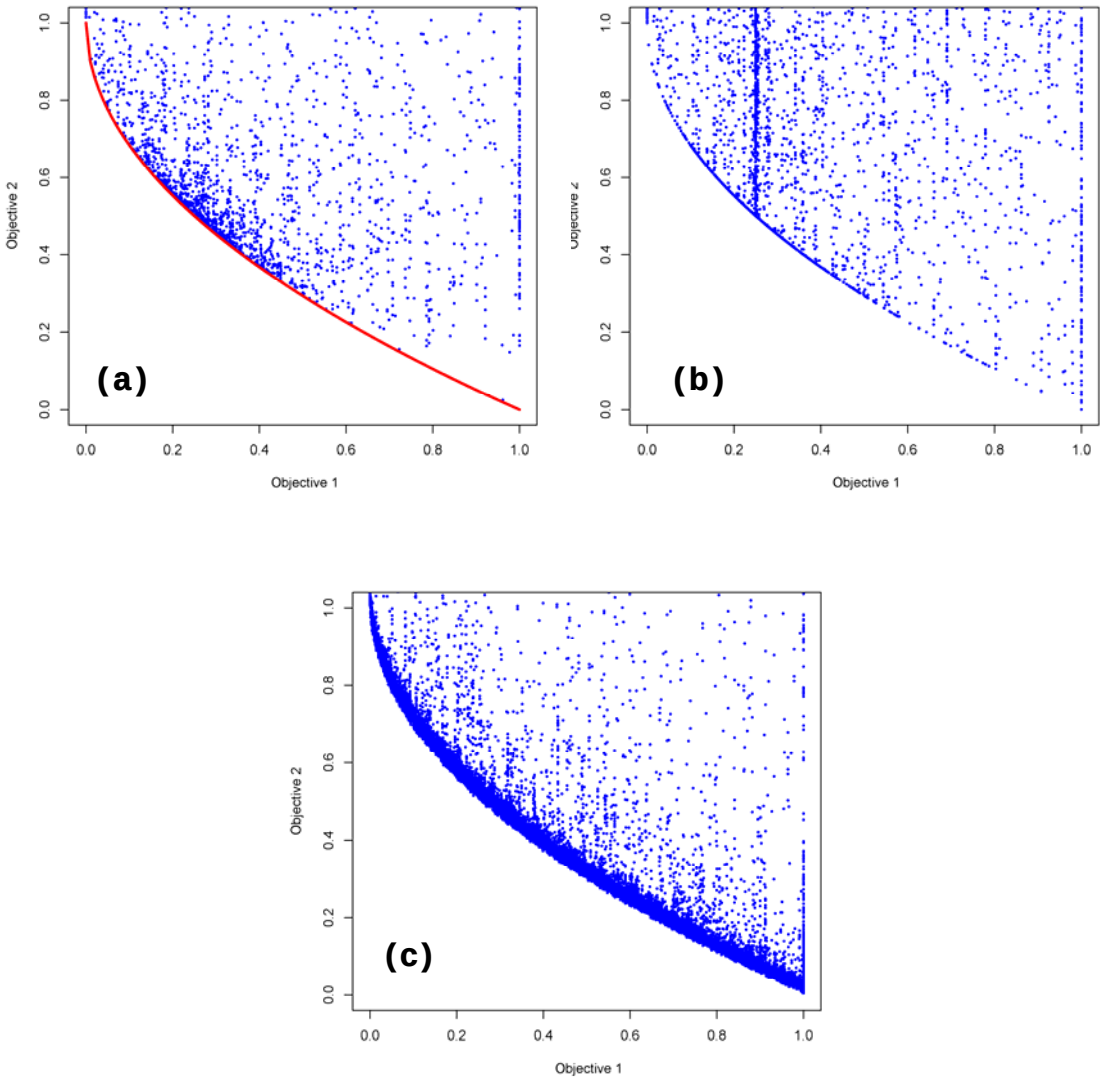

Figure 13: DE-Best results for optimization of the ZDT1 function: (a) using objective sum approach, (b) DEMOPR-Best-1 using Pareto rank × objective sum, and (c) DEMOPR-Best-2 using Pareto ranks only

6.2.2.2 ZDT2

Tables 3 and 4 present the tuning parameters used in DEMOPR and PRACO$_R$ algorithms for ZDT2 test function.

Table 3: Tuning parameters of DEMOPR for random and best strategies in ZDT2 test

Algorithm	N_p	F	C_r	Generations
DE-Rand	100	0.9	0.1	250
DE-Best	100	0.9	0.1	250

Table 4: Tuning parameters of $PRACO_R$ in ZDT2 test

Number of ants	K	q	ξ	Generations
100	100	0.5	0.5	250

As we can see in figure 14, ZDT2 seems to be a difficult benchmark test function and most of the methods have certain problems in converging to the true Pareto front or covering the front with diverse solutions (areas marked by red boxes).

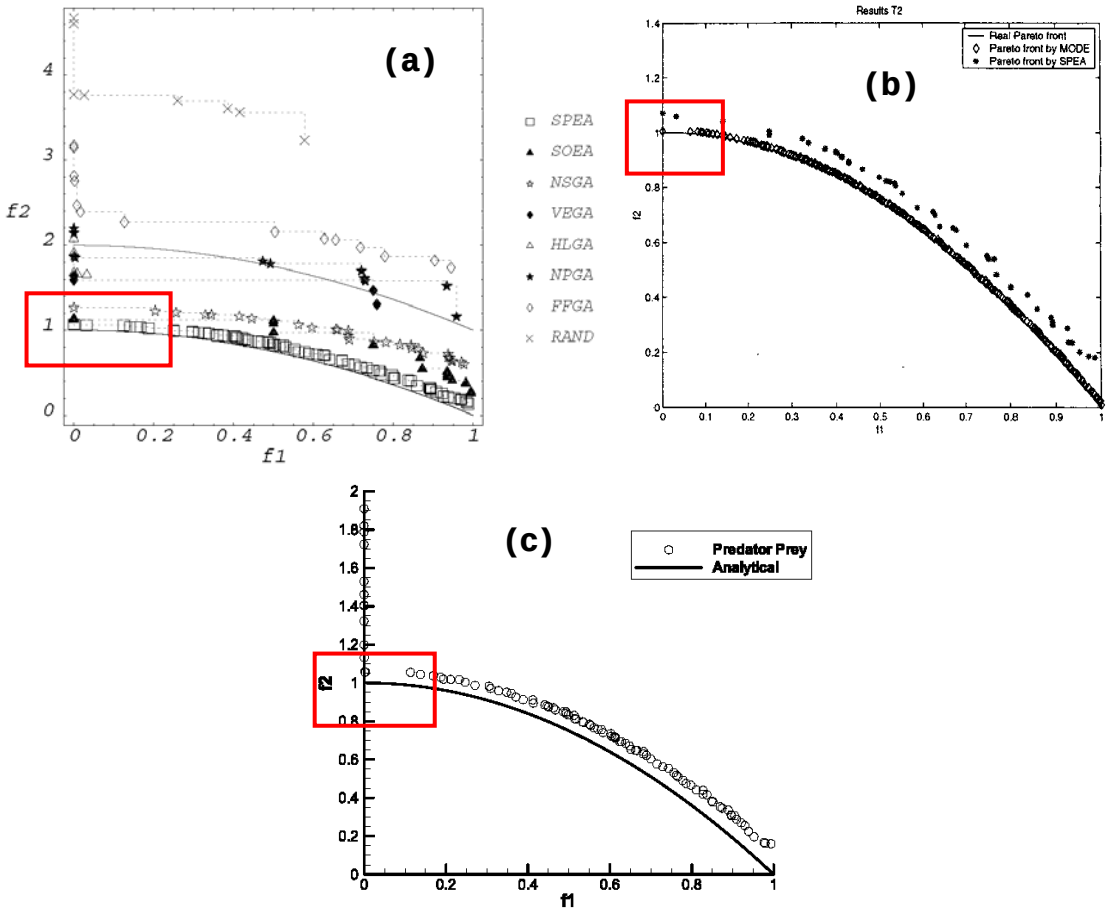

Figure 14: Comparison of different multiobjective algorithms for the ZDT2 test function. (a) after Zitzler et al. [2000], (b) after Xue et al. [2003] and (c) after Chowdhury et al. [2009]

Figure 15 presents the results of the $PRACO_R$ algorithm for the ZDT2 test function. For this test function, neither the objective sum approach nor the $PRACO_R$ method were able to find solutions close to the Pareto front.

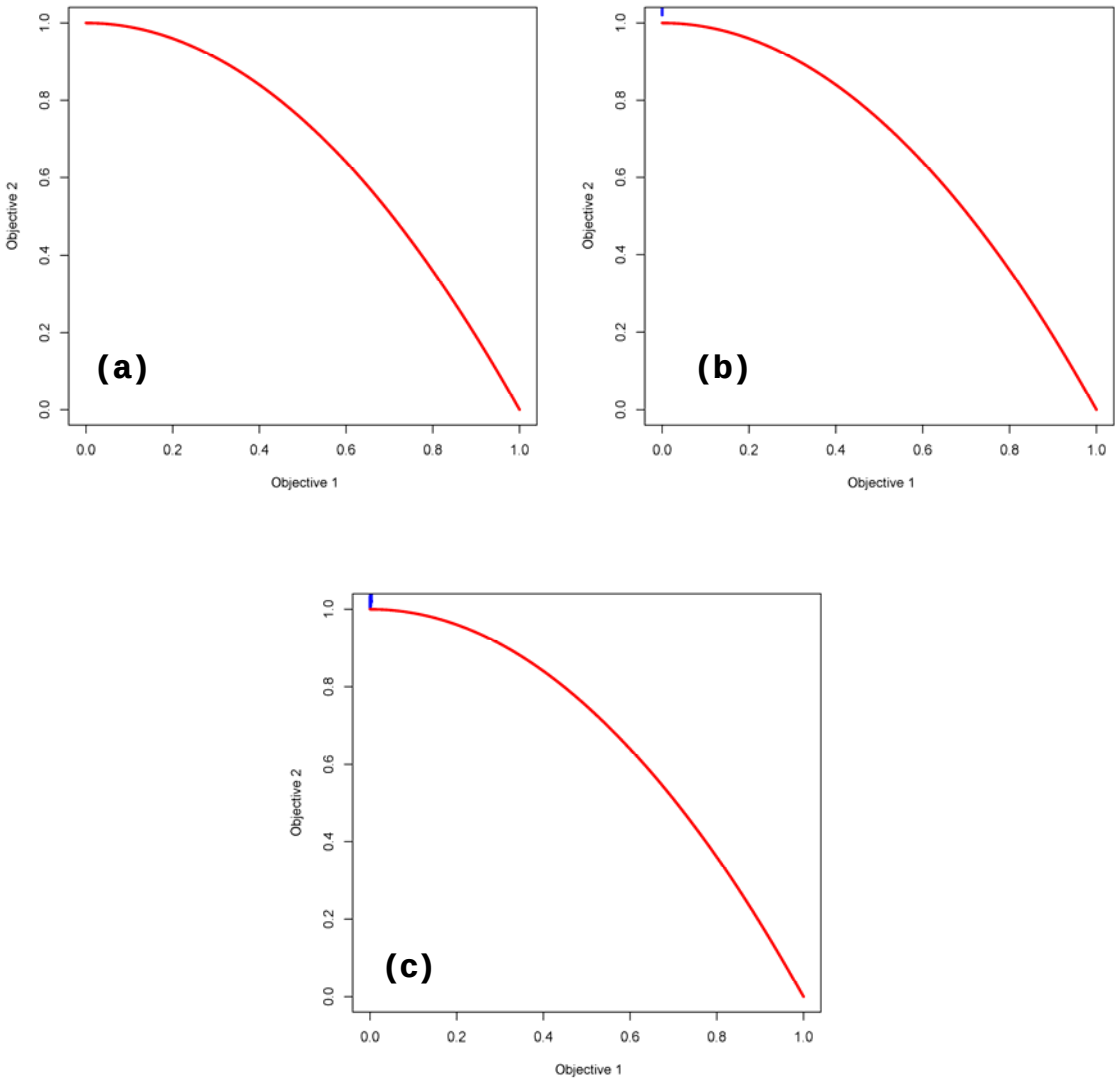

Figure 15: ACO_R results for optimization of the ZDT2 function: (a) using objective sum approach, (b) $PRACO_R$-1 using Pareto rank × objective sum, and (c) $PRACO_R$-2 using Pareto ranks only

We show the results of the DEMOPR-Rand algorithm in figure 16 for the ZDT2 test function. As we see in this figure, only the DEMOPR-Rand-2 was able to find solutions which are located on the whole Pareto front of this test function.

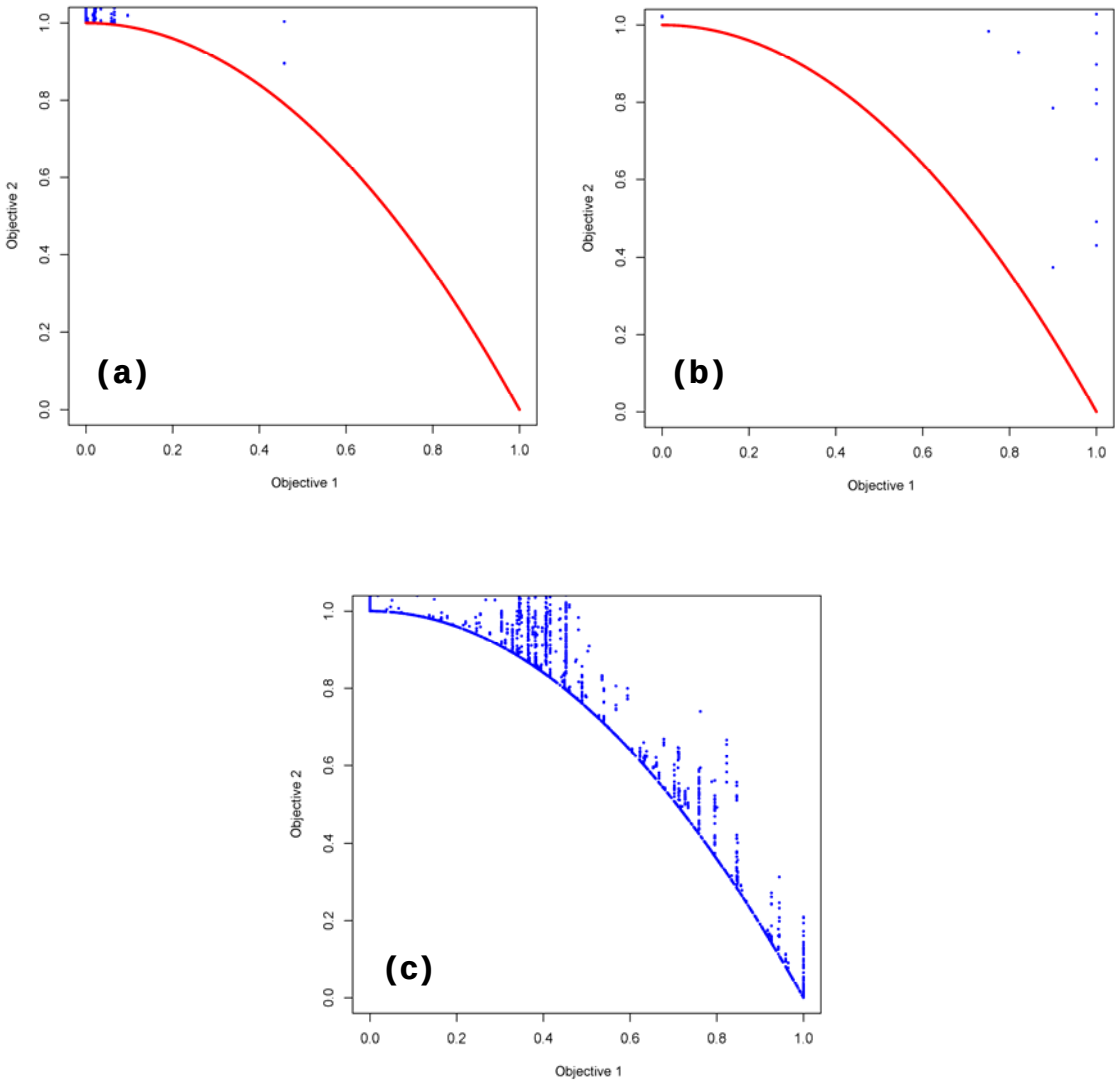

Figure 16: DE-Rand results for optimization of the ZDT2 function: (a) using objective sum approach, (b) DEMOPR-Rand-1 using Pareto rank × objective sum, and (c) DEMOPR-Rand-2 using Pareto ranks only

Figure 17 presents the results of DEMOPR-Best for optimization of the ZDT2 test function. Like the DEMOPR with random strategy, only when we directly use Pareto ranks (DEMOPR-Best-2), we are able to capture the true Pareto front.

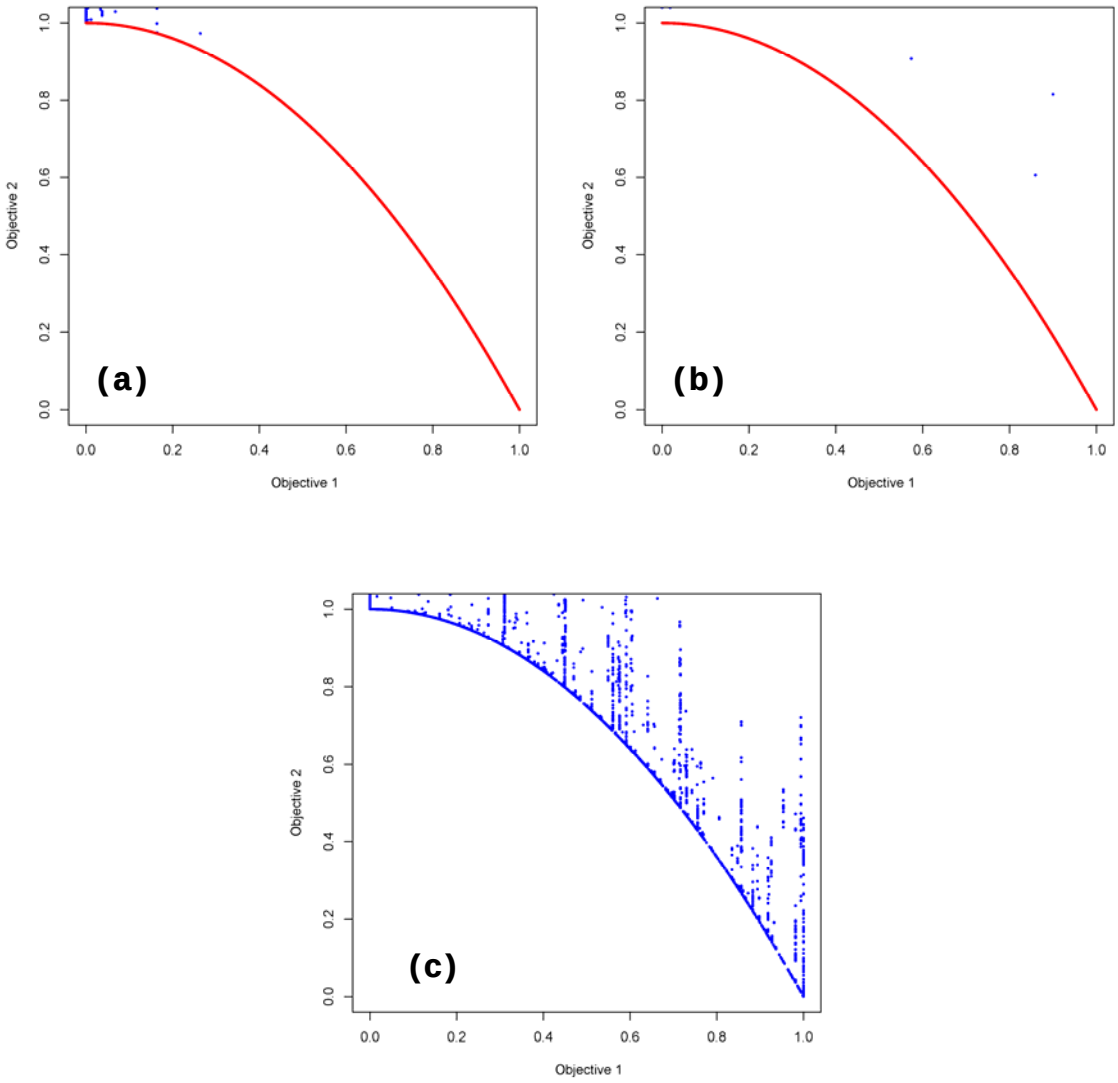

Figure 17: DE-Best results for optimization of the ZDT2 function: (a) using objective sum approach, (b) DEMOPR-Best-1 using Pareto rank × objective sum, and (c) DEMOPR-Best-2 using Pareto ranks only

6.2.2.3 ZDT3

For the ZDT3 test function, we present the tuning parameters used for the algorithms in tables 5 and 6.

Table 5: Tuning parameters of DEMOPR for random and best strategies in ZDT3 test

Algorithm	N_p	F	C_r	Generations
DE-Rand	100	0.5	0.3	250
DE-Best	100	0.3	0.3	250

Table 6: Tuning parameters of $PRACO_R$ in ZDT3 test

Number of ants	k	q	ξ	Generations
100	100	0.3	0.7	250

Figure 18 shows the results of different multiobjective algorithms used for optimization of the ZDT3 benchmark test in order to compare them with our results.

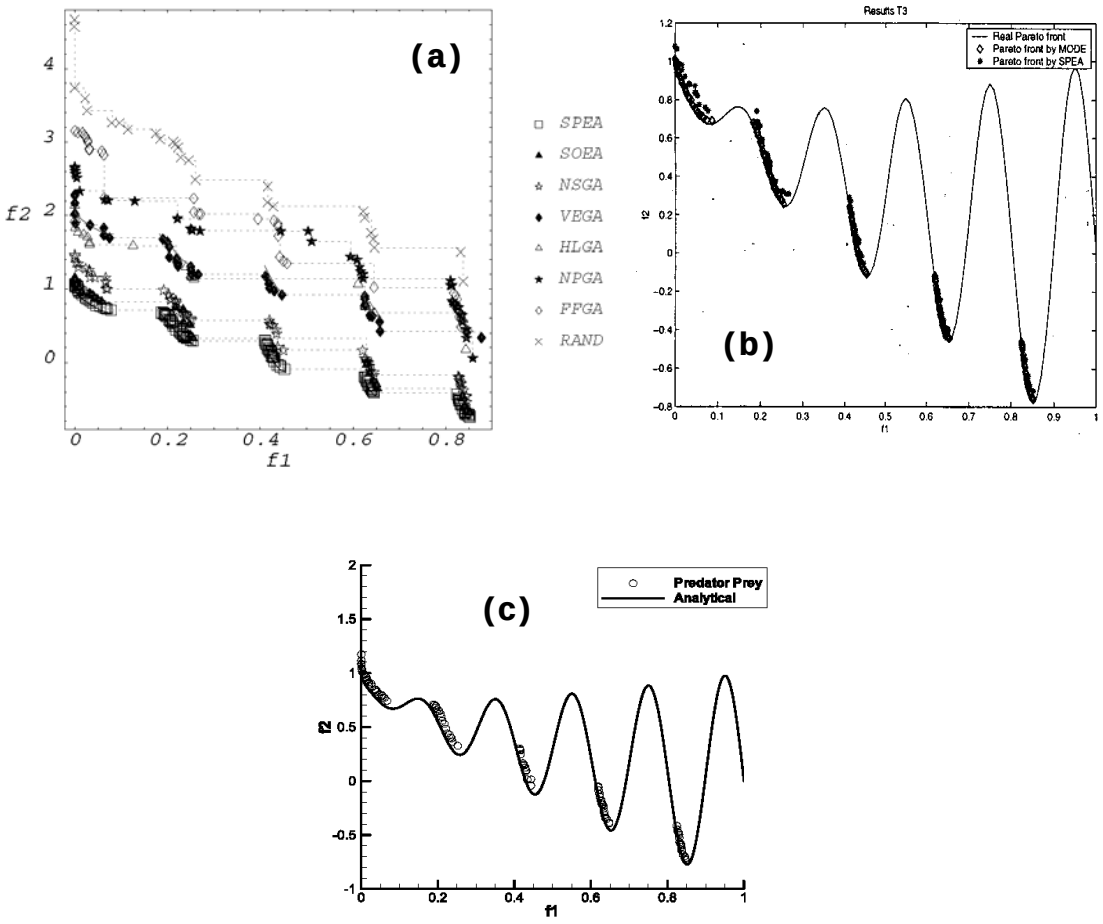

Figure 18: Comparison of different multiobjective algorithms for the ZDT3 test function. (a) after Zitzler et al. [2000], (b) after Xue et al. [2003] and (c) after Chowdhury et al. [2009]

Figure 19 shows the performance of the $PRACO_R$ algorithm for the ZDT3 test function. ZDT3 has several disconnected Pareto fronts. As can be seen in figure 19, using the objective sum approach only leads to discovering one of these Pareto fronts. In $PRACO_R$-1, where we multiply the sum of the objective functions and the Pareto ranks, another swarm of solutions approach the second part of Pareto front as well as covering the first section of the Pareto front.

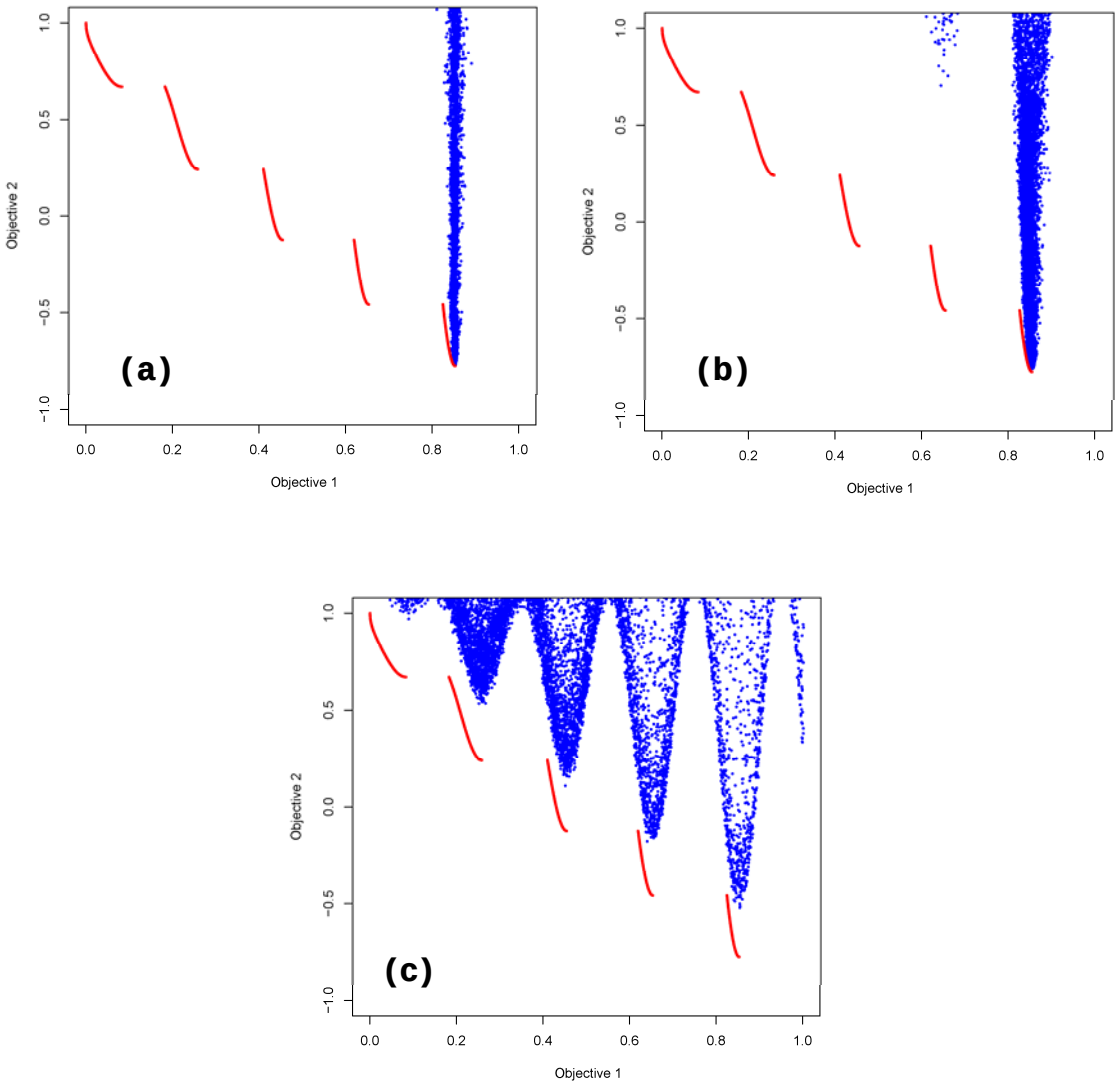

Figure 19: ACO_R results for optimization of the ZDT3 function: (a) using objective sum approach, (b) $PRACO_R$-1 using Pareto rank × objective sum, and (c) $PRACO_R$-2 using Pareto ranks only

Better results are obtained when using the $PRACO_R$-2 algorithm. For this case (figure 19-c), all of the Pareto fronts are discovered. However, the algorithm suffers from an inability to reach to the Pareto front and leaves a gap between obtained solutions and the Pareto front.

In figure 20, we present the results of DEMOPR with random strategy for the ZDT3 test function. Here, we also see the same behavior. Using an objective sum approach, the algorithm fails to recognize all possible Pareto fronts. We obtain better results by

incorporating Pareto ranks with the objective sum approach. The best results are obtained by using DEMOPR-Rand-2, where we rely only on Pareto ranks in differential evolution optimization. We are not only able to identify all disconnected Pareto fronts, but also we obtain solutions on the Pareto front, with no gap between the set of solutions produced by DEMOPR-Rand-2 and multiple Pareto fronts.

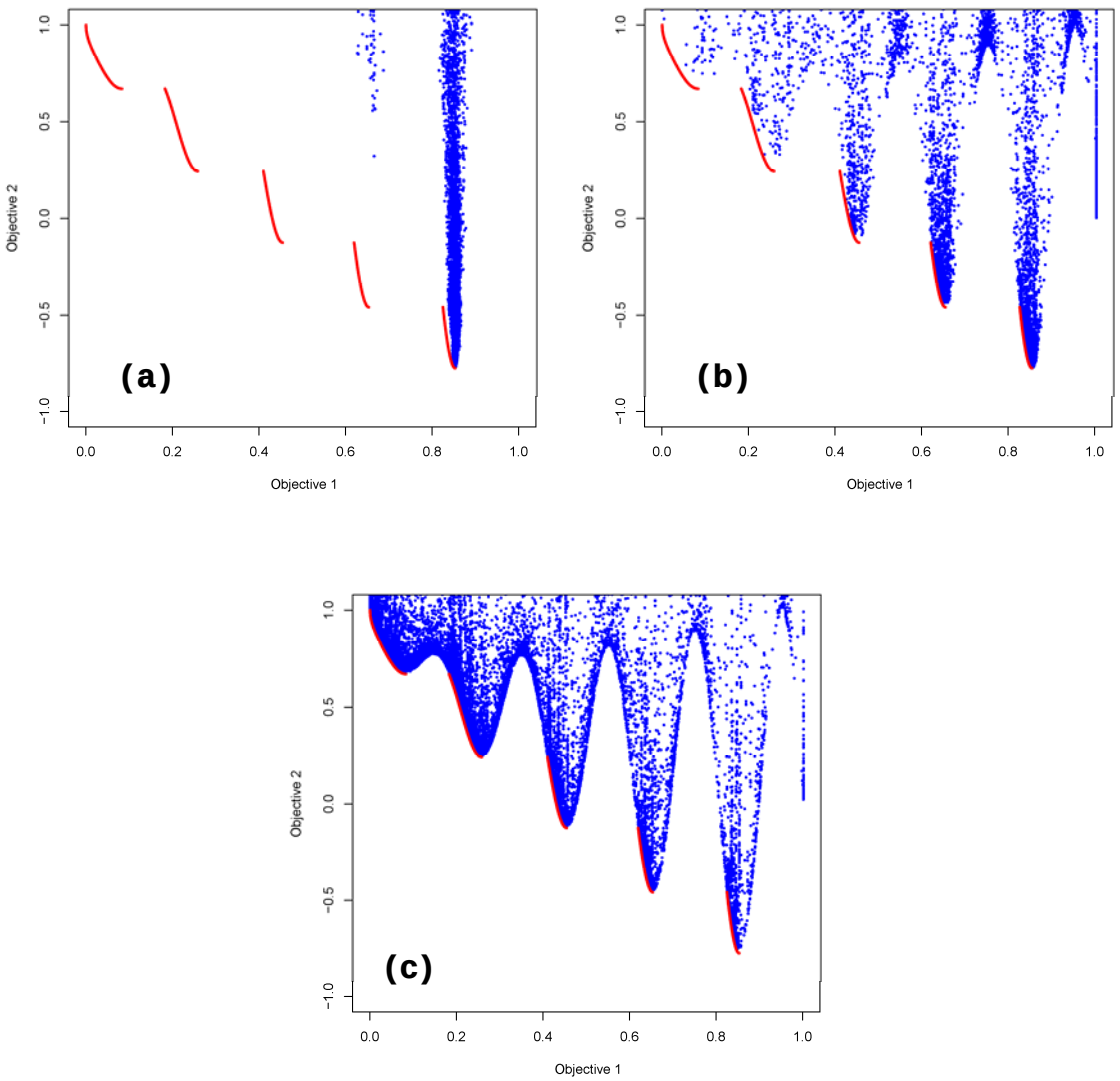

Figure 20: DE-Rand results for optimization of the ZDT3 function: (a) using objective sum approach, (b) DEMOPR-Rand-1 using Pareto rank × objective sum, and (c) DEMOPR-Rand-2 using Pareto ranks only

Figure 21 shows the results of the DEMOPR-Best algorithm. As we expect, DEMOPR-Best-2 gives better results than (a) the objective function sum and (b) the objective

functions multiplied by Pareto ranks. However, the DEMOPR-Best is not able to get close to all of the disconnected fronts.

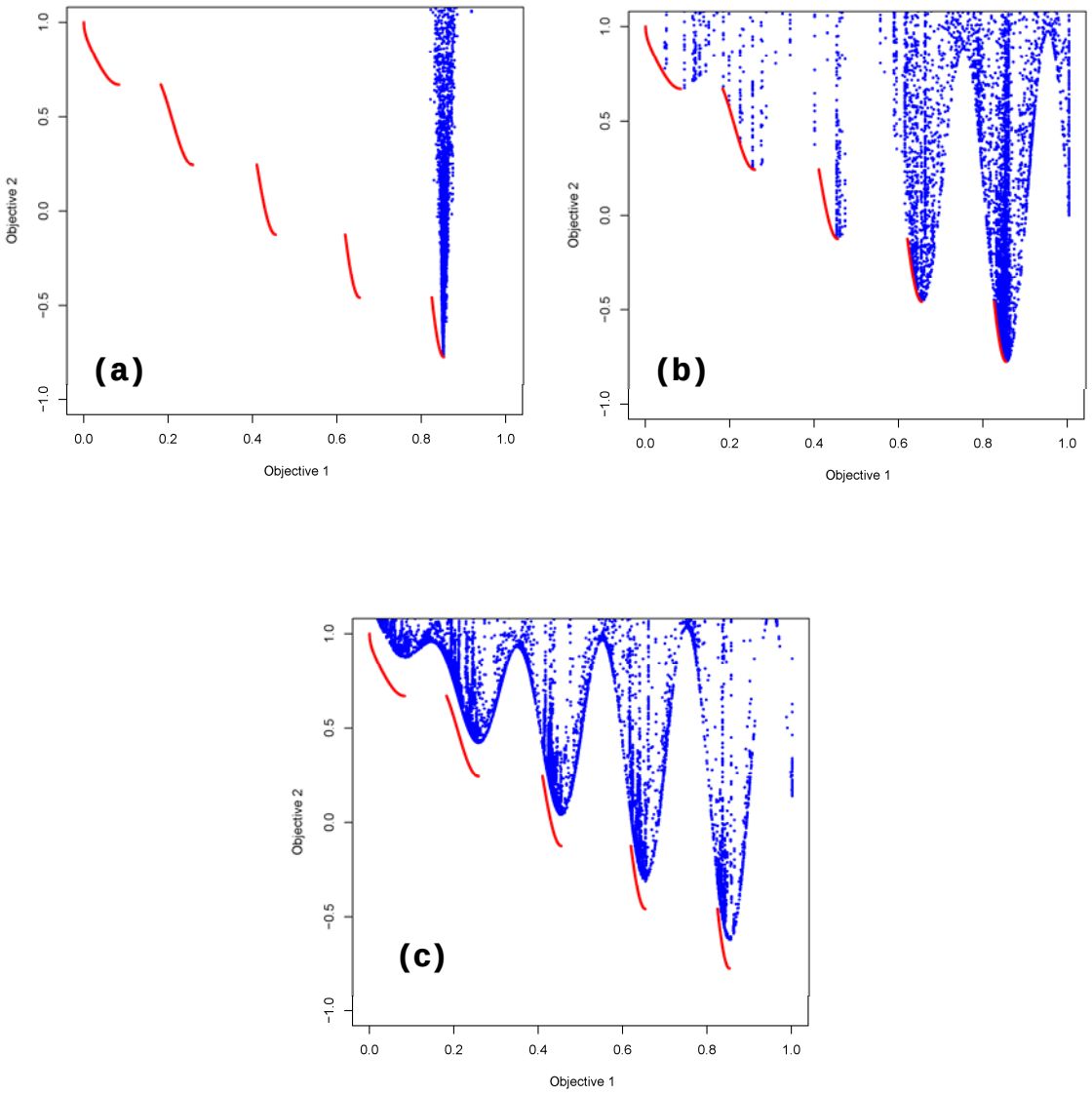

Figure 21: DE-Best results for optimization of the ZDT3 function: (a) using objective sum approach, (b) DEMOPR-Best-1 using Pareto rank × objective sum, and (c) DEMOPR-Best-2 using Pareto ranks only

To conclude, using Pareto rank approach (PRACO$_R$-2 and DEMOPR-2) always results in a better performance of the algorithms than using the objective sum and quick-fix approaches (PRACO$_R$-1 and DEMOPR-1). DEMOPR-Rand for these test functions has shown a better performance than PRACO$_R$ and DEMOPR-Best algorithms. We select

DEMOPR-Rand for multiobjective history matching and present the results in the next section.

It should be noted that in this section, we only rely on a visual comparison of algorithms in obtaining close and diverse solutions on the Pareto front. For a better comparison between proposed algorithms, robust statistical tests and performance metrics such as those introduce by Zitzler et al. [2000] may be performed on the results. It is also desirable to conduct a full parameter study in order to gain more insight into the sensitivity of the algorithms to tuning parameters.

6.3 Multiobjective History Matching: PUNQ-S3 Reservoir

After testing the developed multiobjective algorithms on three benchmark functions, we selected the DEMOPR algorithm with the "random" strategy for multiobjective history matching. The PUNQ-S3 model was used in this chapter to test the performance of this algorithm for multiobjective history matching and, by coupling with a Bayesian framework, for uncertainty quantification.

To remind the setup of PUNQ-S3 model used before, we briefly present the parameterization and initial ranges of the parameters in this section. The PUNQ-S3 model in this study has five layers and nine homogenous regions per layer. The DEMOPR algorithm was used to estimate porosities in each homogenous region and layer in the reservoir. Five layers times nine regions per layer makes 45 porosity values that are estimated in this problem. Equations 10 and 11 are used to estimate horizontal and vertical permeability from porosity values [Boss, 1999]. These correlations are obtained based on least square fitting of the well data crossplots.

$$\ln(k_h) = 0.77 + 9.03\phi \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (10)$$

$$k_v = 3.124 + 0.306k_h \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (11)$$

The initial ranges for unknown parameters in each layer are given in table 7.

Table 7: Initial ranges for unknown parameters in PUNQ-S3 reservoir

Layer	Porosity	Horizontal Permeability (md)	Vertical permeability (md)
1	0.15 - 0.3	133 - 3013	44 - 925
2	0.05 – 0.15	16 - 133	8 - 44
3	0.15 - 0.3	133 - 3013	44 - 925
4	0.1 - 0.2	47 - 376	17 - 118
5	0.15 - 0.3	133 - 3013	44 - 925

The following objective function was minimized using the DEMOPR algorithm [Barker et al. 2001]:

$$SoS(o^{obs}, p) = \frac{1}{n_w}\sum_i \frac{1}{n_p}\sum_j \frac{1}{n_t}\sum_k \left(w_{ijk} \frac{o_{ij}^{obs}(t^k) - o_{ij}^{sim}(t^k; p)}{\sigma_{ijk}} \right)^2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (12)$$

where n_w is number of wells with subscript i running over the wells, n_p is number of production data types with subscript j running over them. Subscript k runs over production data report times and n_t is the respective number of samples. Observed data (o^{obs}) and simulated ones (o^{sim}) for each of the parameters (p) are being reported at time steps t^k with measurement error of σ. At each time step for the parameters, there are extra weighting factors denoted with w. These weights reflect the importance of some of data types at specific time steps and are specifically indicated in the online dataset.

Since the DEMOPR algorithm is designed to handle two objectives, we need to separate different components of the objective function coming from different wells into two groups. It is decided to group wells 1, 4 and 5 as the first objective function and remaining wells (11, 12 and 15) as the second objective. BHP, GOR and WCT of each group are summed up to form the objective function in each group. This, obviously, follows the original definition of the objective function in equation 12. In the next section, first history matching using DEMOPR algorithm is described and then we will investigate the effect of multiobjective history matching on uncertainty estimates.

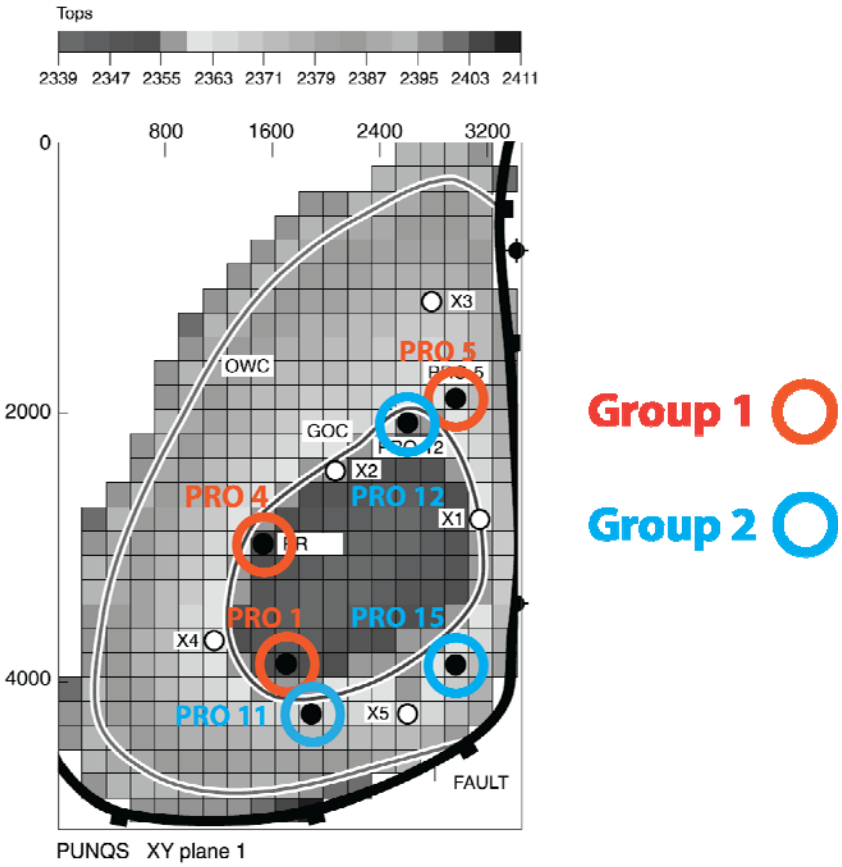

Figure 22: PUNQ-S3 top structure map and grouping of wells

6.3.1 History Matching

For history matching the PUNQ-S3 model with the DEMOPR algorithm, we have used the tuning parameters reported in table 8. In order to examine the performance of the DEMOPR algorithm versus the objective-sum approach, we have performed 10 trials with the same tuning parameters, but different initial random seeds. Objective-sum approach uses the original differential algorithm with "random" strategy and sums up the two objective functions obtained for each well group to obtain the global misfit value.

Table 8: Tuning parameters used for DEMOPR algorithm and objective-sum approach

Population size (N_p)	Scaling factor (F)	Crossover rate (C_r)	Simulations
50	0.3	0.3	3050

Figure 23 presents the best misfit results in 10 trials for both DEMOPR and objective-sum approaches.

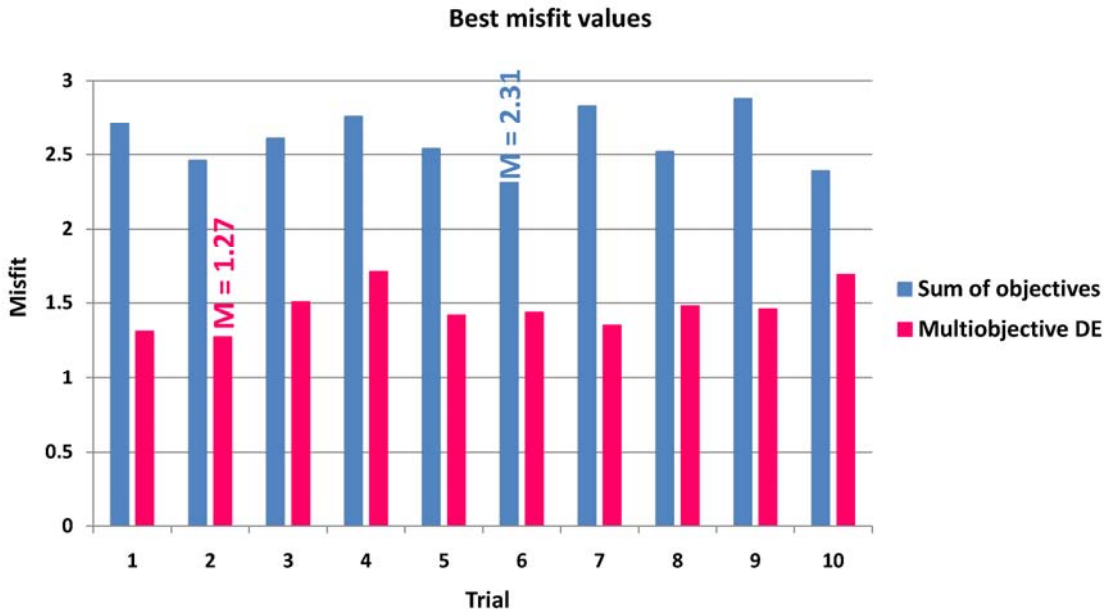

Figure 23: Comparison of best misfit values in 10 trials for objective-sum and DEMOPR algorithm

Examining figure 23, we understand that in all of 10 trials, the DEMOPR algorithm provides a lower objective function value than the traditional objective-sum approach. The minimum misfit value for the objective sum approach is 2.31, while DEMOPR has a best misfit of 1.27 among these 10 runs. We should note that a small scaling factor is selected for this test to get an even faster convergence for both DEMOPR and objective-sum approaches. Remember from chapter 5 that the best misfit value obtained with the same population size and number of simulations, but different scaling factors and crossover rates with DE/Rand algorithm is 1.95. All multiobjective differential evolution trials obtain lower misfit values than this best misfit. The mean misfit for the 10 trials using DEMOPR algorithm is 1.45.

Figure 24 shows boxplots for misfit values based on the same information used to plot figure 23. Examination of the figure suggests that not only does the DEMOPR algorithm obtain lower misfit values for the 10 trials, but also the distance between upper and lower quartiles (IQR) is more compact for multiobjective optimization. This means that when history matching the PUNQ-S3 model, DEMOPR shows less variability for initial seed value and is more robust to randomness in initialization of the optimization in comparison with the objective-sum approach.

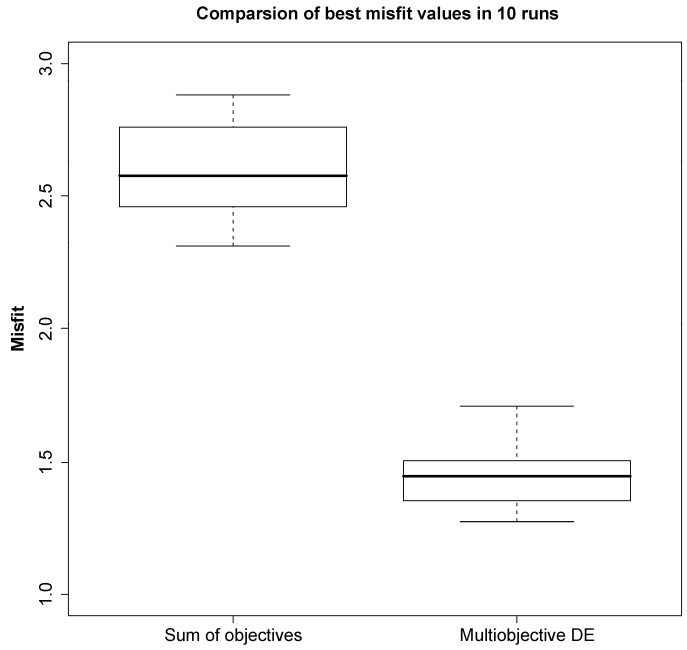

Figure 24: Boxplots of minimum misfits obtained for objective-sum and DEMOPR algorithms

In multiobjective optimization, we are not only interested in obtaining better models, but also we want a wider distribution and coverage of the Pareto front. For examining this, we show the spread of the two objective function values in figure 25 for the multiobjective DE and objective-sum approaches.

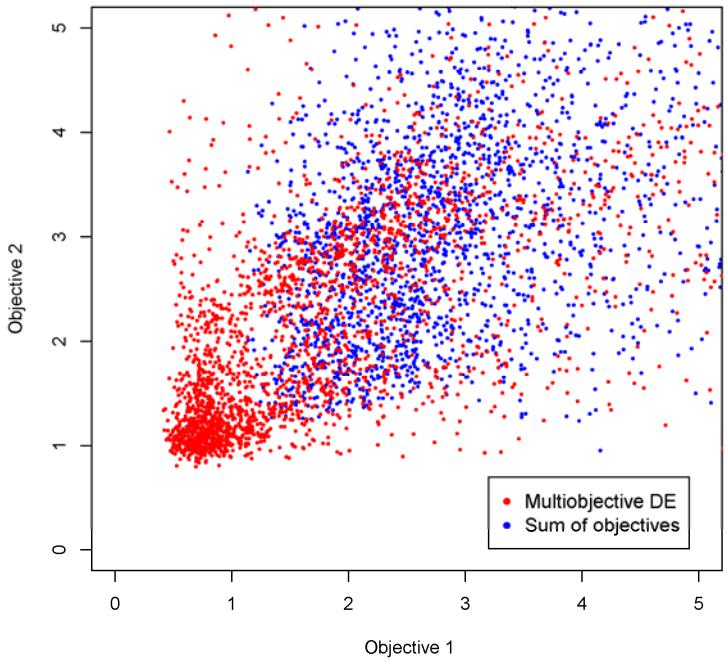

Figure 25: Spread of models in objective function space for objective sum and multiobjective DE approaches

Figure 25 suggests that the DEMOPR algorithm obtains a closer front of solutions towards the Pareto front. Also models obtained by DEMOPR have a wider spread in objective function space.

After comparing the best misfits for the DEMOPR algorithm and objective-sum approaches, we may also investigate the convergence speeds of both approaches. Figure 26 compares the best misfit values in each generation for these two approaches. Dark blue and red lines represent the global objective function values for objective-sum and DEMOPR algorithms. Orange and pink lines show misfit values for two well groups in the DEMOPR algorithm. Light blue and purple lines indicate these two misfit groups for the objective-sum approach using DE algorithm.

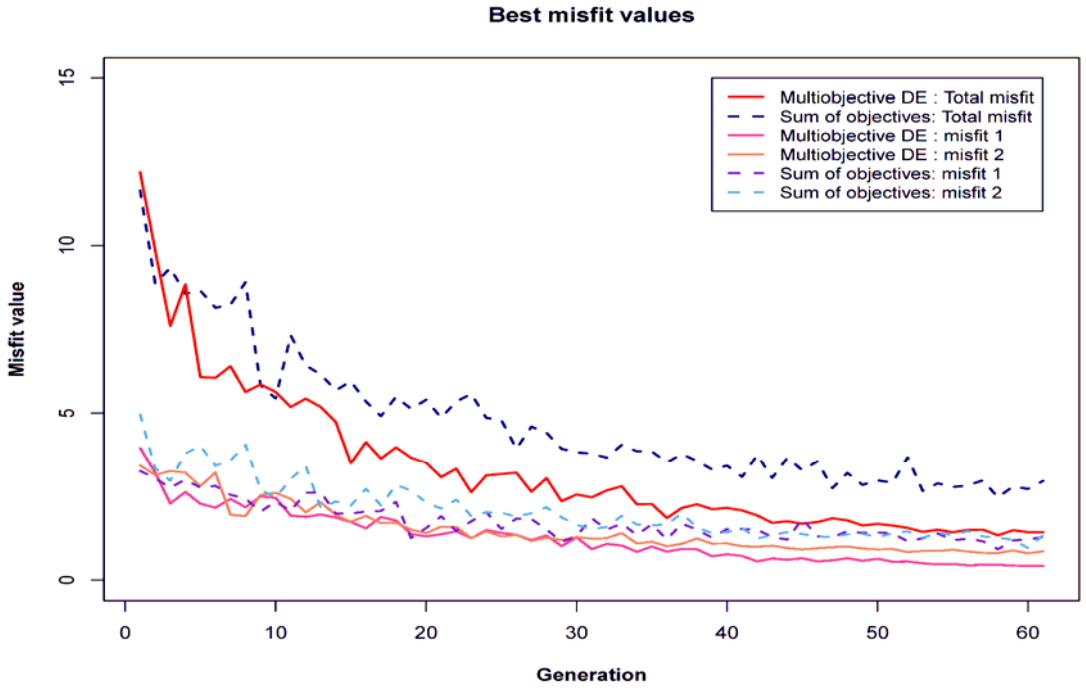

Figure 26: Comparison of misfit reduction efficiency for objective-sum and DEMOPR approaches

The global objective function is obtained by summing up the misfit values of the two well groups in the DEMOPR and objective-sum approaches. Figure 26 shows that both approaches start from almost same misfit value but the DEMOPR algorithm reduces the misfit value faster during optimization and ends up with a lower misfit value in comparison with the objective-sum approach. This observation also agrees with

individual components of the global objective function (i.e. misfit values coming from two well groups).

Further to a comparison of the best misfit values in each generation, we can also have a look at generational misfit values and compare DEMOPR and objective-sum approaches. Figure 27 shows the boxplots for two approaches. Each generation consists of 50 individuals. Two different zoom levels are provided for better visualization and comparison of DEMOPR algorithm vs. objective sum DE.

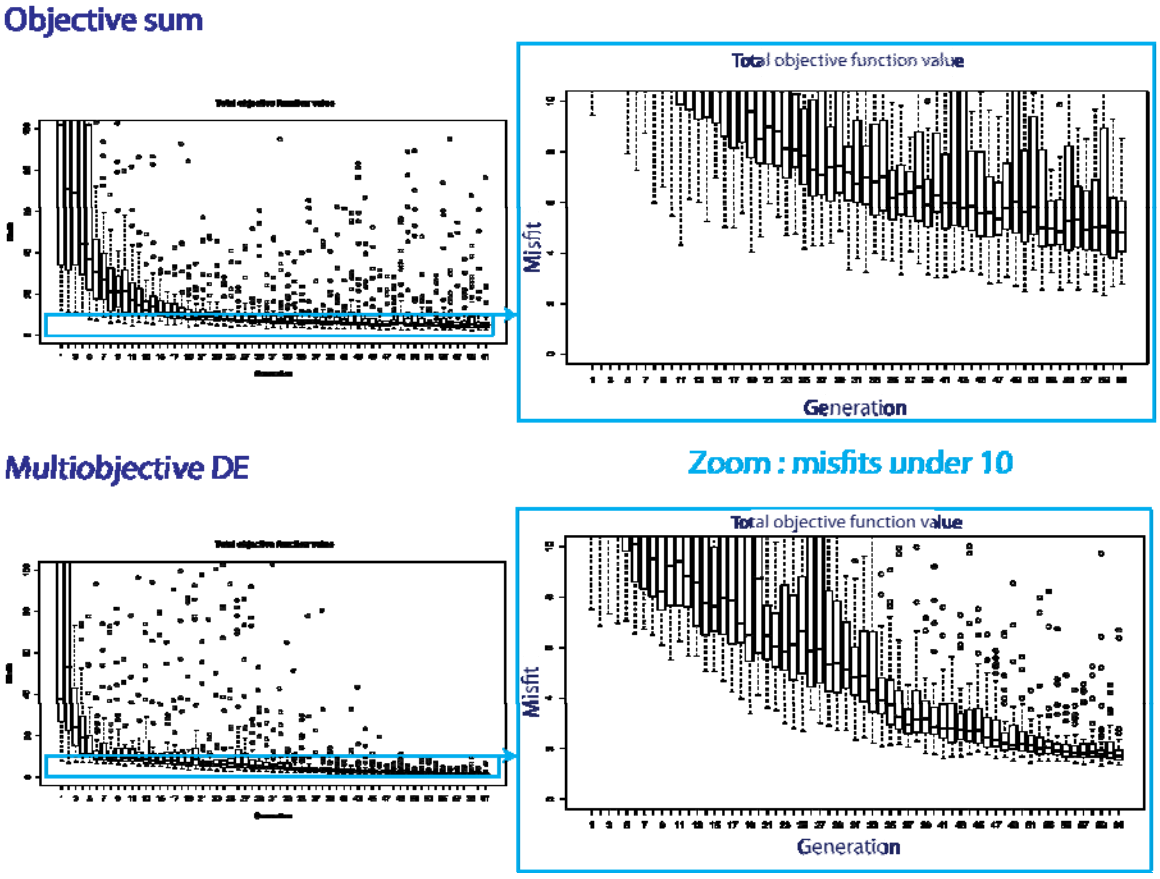

Figure 27: Boxplots showing generational statistics for objective-sum and multiobjective DE approaches

Based on figure 27, we conclude that the DEMOPR algorithm is also more efficient in terms of the generational misfit reduction. Both approaches, as stated in table 8, use tuning parameters that result in a fast convergence. The traditional objective-sum approach using differential evolution algorithm suffers from larger difference between lower and upper quartiles of boxplots in each generation (wide IQR). This means, even

at final generations, there is a significant difference between misfits obtained and many poor history-matched models are also generated beside some good ones. In contrast, DEMOPR finds more good history-matched models during the sampling process. This, in practice, translates to compact misfit boxplots in each generation in the DEMOPR algorithm.

We can also visualize the efficiency of the DEMOPR algorithm in reducing misfit values through another lens. We can use the Hierarchical Clustering Explorer (HCE) visualization toolkit developed by Human-Computer Interaction Lab at University of Maryland [Seo and Shneiderman, 2005] for this purpose. Figure 28 presents the heatmap visualizations of the two objective function values for objective-sum and multiobjective differential evolution optimization. In each approach, parallel horizontal bars represent two misfit values coming from well groups. On the horizontal axis, we have the 3050 models obtained during history matching. Each colored vertical bar represents one model. The color of the bar shows misfit value, with blue colors showing lower objective functions.

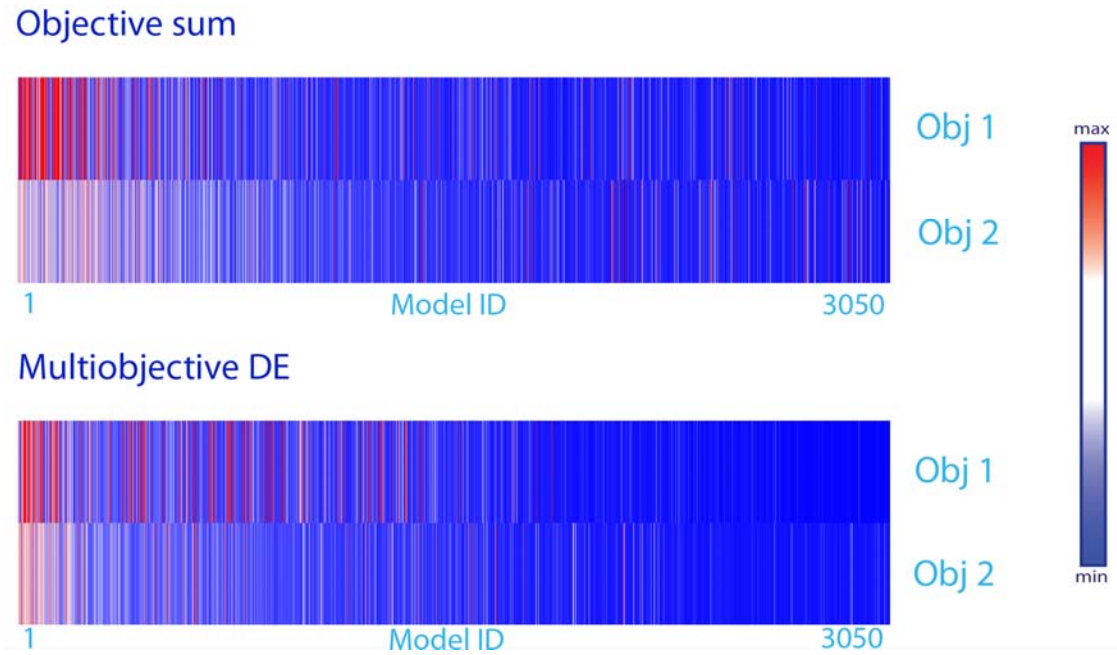

Figure 28: Heatmap visualization of two objective functions for objective-sum and multiobjective DE approaches

We can see that for the objective-sum approach, we still get a mixture of red and blue color bars in the final stages of history matching. Appearance of high misfit values (red bars) during final iterations means that the objective-sum approach is not very efficient in misfit reduction for this multiobjective history matching problem. The contrast of colors is clearly less for the DEMOPR algorithm. In the final stages of history matching using DEMOPR we observe more smooth bars of blue, which means that both misfit values for well groups have been reduced effectively.

Based on the results presented in this section, we have demonstrated the superiority of the multiobjective DE algorithm for history matching in comparison with the traditional objective-sum approach.. After comparing both approaches for history matching, we shall now examine the effect of the proposed framework on uncertainty estimates of future reservoir performance.

6.3.2 Uncertainty Quantification

After obtaining history-matched models using the DEMOPR algorithm, we can proceed to uncertainty quantification of the predictions. The complete ensemble of models in each trial is submitted to the NAB algorithm to calculate Bayesian credible intervals. Figures 29 and 30 present the uncertainty of total oil recovery from PUNQ-S3 after 16.5 years in each of the 10 trials using the objective-sum and DEMOPR algorithms.

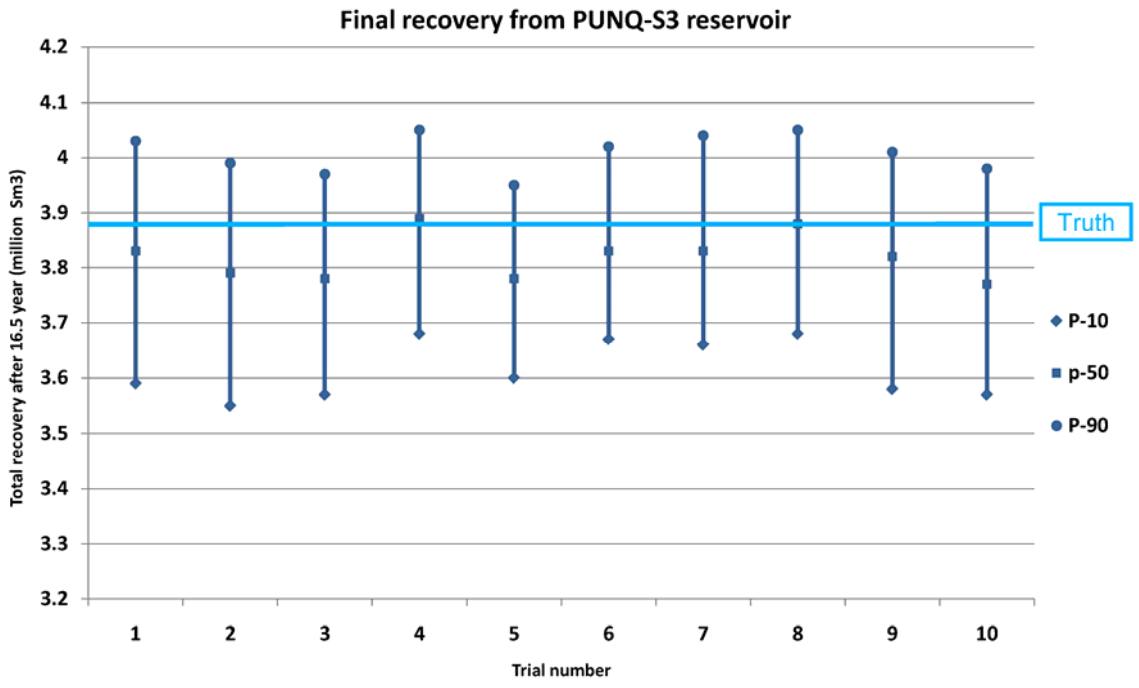

Figure 29: Uncertainty intervals obtained by objective-sum approach

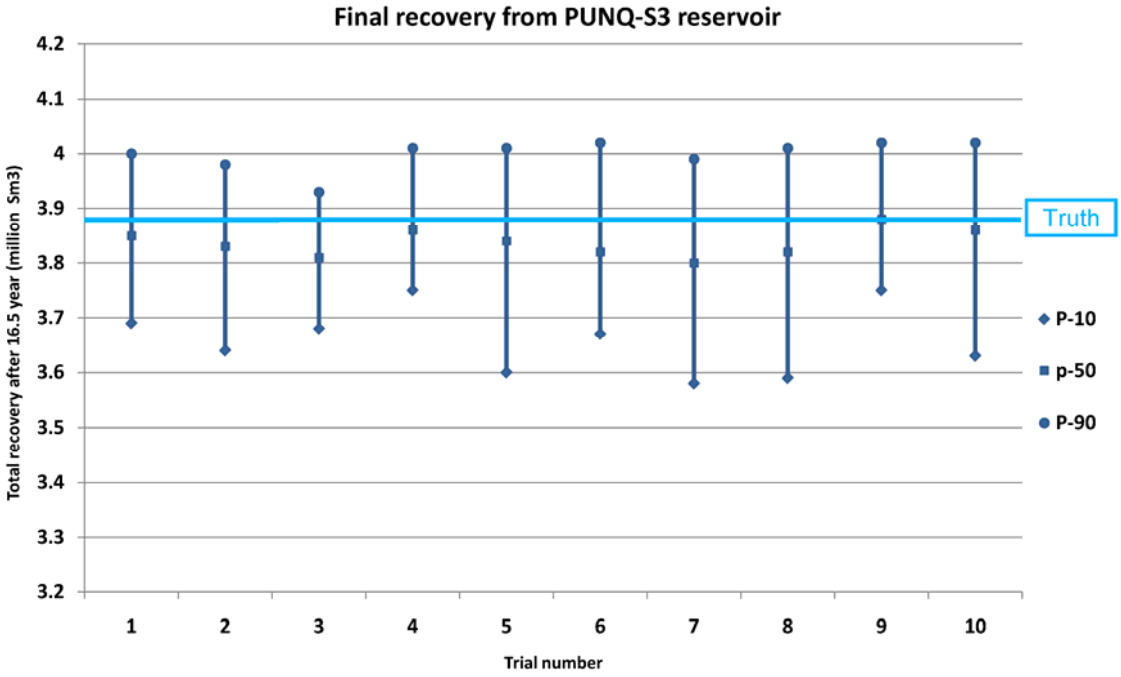

Figure 30: Uncertainty intervals obtained by multiobjective DE approach

To summarize the results of figures 29 and 30, we have presented the average and standard deviation of P10-P50-P90 estimates of ultimate oil recovery in 10 runs using the objective sum and multiobjective DE algorithms in table 9.

Table 9: Average and standard deviation of Bayesian credible intervals for total oil recovery in objective sum and multiobjective approaches (units: million Sm^3)

Approach	P10	P50	P90
Objective sum	3.61 ± 0.06	3.82 ± 0.04	4.01 ± 0.3
DEMOPR	3.66 ± 0.05	3.84 ± 0.02	3.99 ± 0.2

Examining figures 29 and 30 and table 9, we see that the multiobjective differential evolution algorithm (DEMOPR), on average, provides a closer P50 value to the truth solution (3.87 million Sm^3). Also the difference between the P10 and P90 estimates is less for the DEMOPR algorithm than for the objective-sum differential evolution.

One of the key questions that should be addressed in uncertainty quantification is the number of models needed in history matching to obtain a reliable and stable uncertainty

estimate. The practical importance of this question is related to the computational resources required to perform many forward reservoir simulations. In a typical assisted history matching framework, multiple history-matched models are generated and the resulting ensemble of models is used for inference and uncertainty quantification. Any method that can provide stable and reliable uncertainty estimates from a smaller ensemble size can greatly help to minimize computational costs.

To address this question, we performed a test with different numbers of models in the ensemble to assess the impact of multiobjective history matching. The 10 trials introduced earlier in this chapter for the objective-sum and DEMOPR algorithms are selected for performing this study. We split each complete ensemble of 3050 models into different parts and submit the models to NAB algorithm. For example we used the first 50 models of the complete ensemble to form the first sub-ensemble. We have selected the following number of models for forming each sub-ensemble: 50, 100, 150, 200, 250, 300, 600, 900, 1200, 1500, 1800, 2100, 2400, 2700 and 3050. For each sub-ensemble, we performed one NAB run and recorded the P10, P50 and P90 estimates. This process was repeated for all of the sub-ensembles for 10 trials. Then the Bayesian credible intervals obtained for each sub-ensemble were averaged based on 10 runs.

Figure 31 presents the results of this study. Warm colors (red, pink and orange) show the multiobjective differential evolution algorithm (DEMOPR) and cold colors (dark blue, purple and light blue) show the objective-sum DE algorithm. This figure shows that, on average, DEMOPR needed fewer models to obtain a stable uncertainty estimate. For example, considering the P50 estimate, we see that the DEMOPR algorithm obtains a stable line using 600 models, but in the objective-sum DE approach, we needed to use at least 1200 models for this purpose.

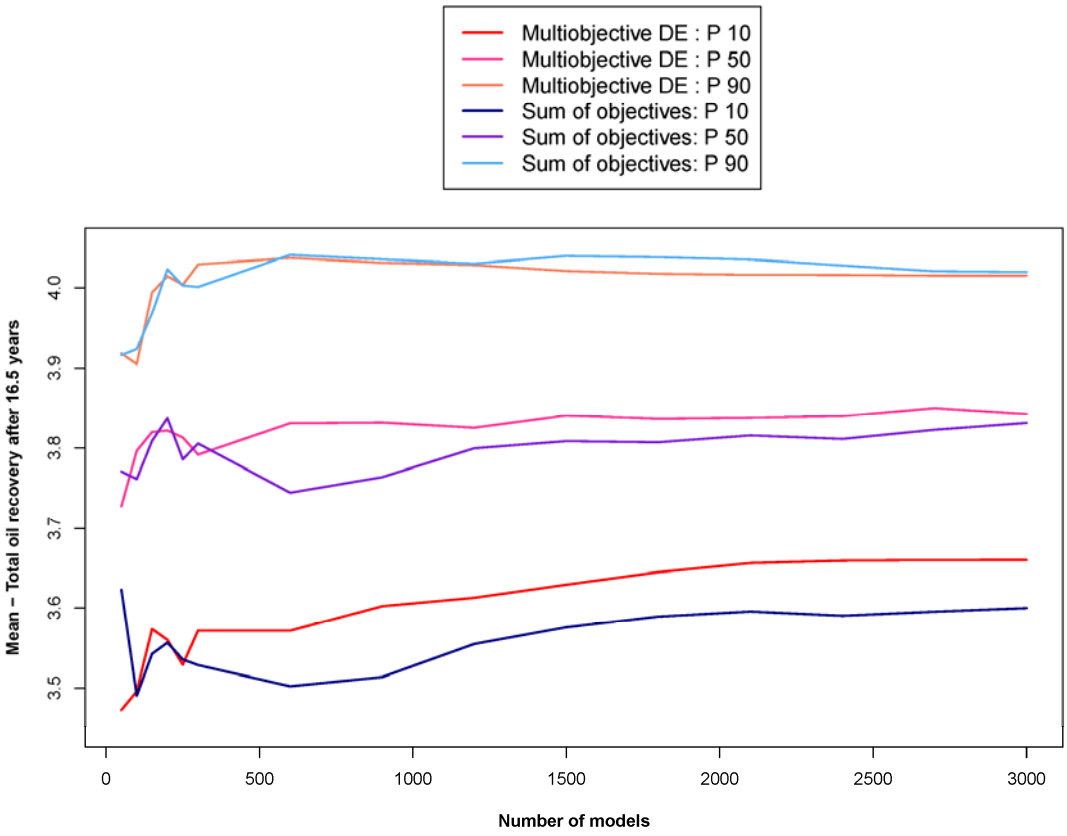

Figure 31: Stabilization of Bayesian credible intervals (P10, P50, P90) for objective sum and objective sum approached (obtained by averaging of 10 runs)

To summarize this chapter, we conclude that the multiobjective differential evolution has been shown to be superior to the conventional objective-sum approach both in history matching and uncertainty quantification. The DEMOPR algorithm enjoys a faster convergence during history matching in comparison with the standard DE. Generational misfits are also lower for DEMOPR algorithm. This new approach also provides more accurate uncertainty estimates for the case study investigated in this chapter. Finally it has been shown that, using multiobjective differential evolution, we need fewer simulations to obtain stable Bayesian credible intervals. The results of this chapter show the benefits of multiobjective optimization in history matching and uncertainty quantification and suggest that this approach should be considered as an important improvement to current assisted history matching workflows.

Chapter 7

> "Reasoning draws a conclusion, but does not make the conclusion certain, unless the mind discovers it by the path of experience"
>
> Roger Bacon, English Philosopher (1214-1294)

Conclusions and Recommendations

The main concern of this thesis was to evaluate the latest developments in computer science such as ant colony optimization, differential evolution and multiobjective optimization in the context of improving history matching and uncertainty quantification. This chapter summarizes the contributions of this thesis, its key findings and recommendations for the future research directions.

7.1 Major Contributions

The research conducted throughout this study investigated the application of two novel optimization algorithms for generating an ensemble of history matched models and quantifying the uncertainty in the predictions made by this ensemble. In this context, the major contributions of current work can be summarized as:

i. A formal and coherent literature review on history matching, uncertainty quantification and multi-agent optimization algorithms
ii. Application of ant colony optimization to history matching
iii. Application of differential evolution to history matching
iv. Comparison of these algorithms with the neighbourhood algorithm

v. Studying the effect of production data on performance of multi-agent optimization algorithms in making reservoir production forecasts

vi. Development and application of multiobjective versions of ant colony optimization and differential evolution algorithms for history matching and uncertainty quantification

7.2 Key Findings

Based on the work performed in this project, the following points can be considered as conclusions of the research:

By bringing ant colony optimization (ACO_R), differential evolution (DE) and multiobjective optimization to the arena of history matching and uncertainty quantification, we have improved current frameworks for these areas. Figure 1 presents a comparison of these new developments with the neighbourhood algorithm as the benchmark method in this study.

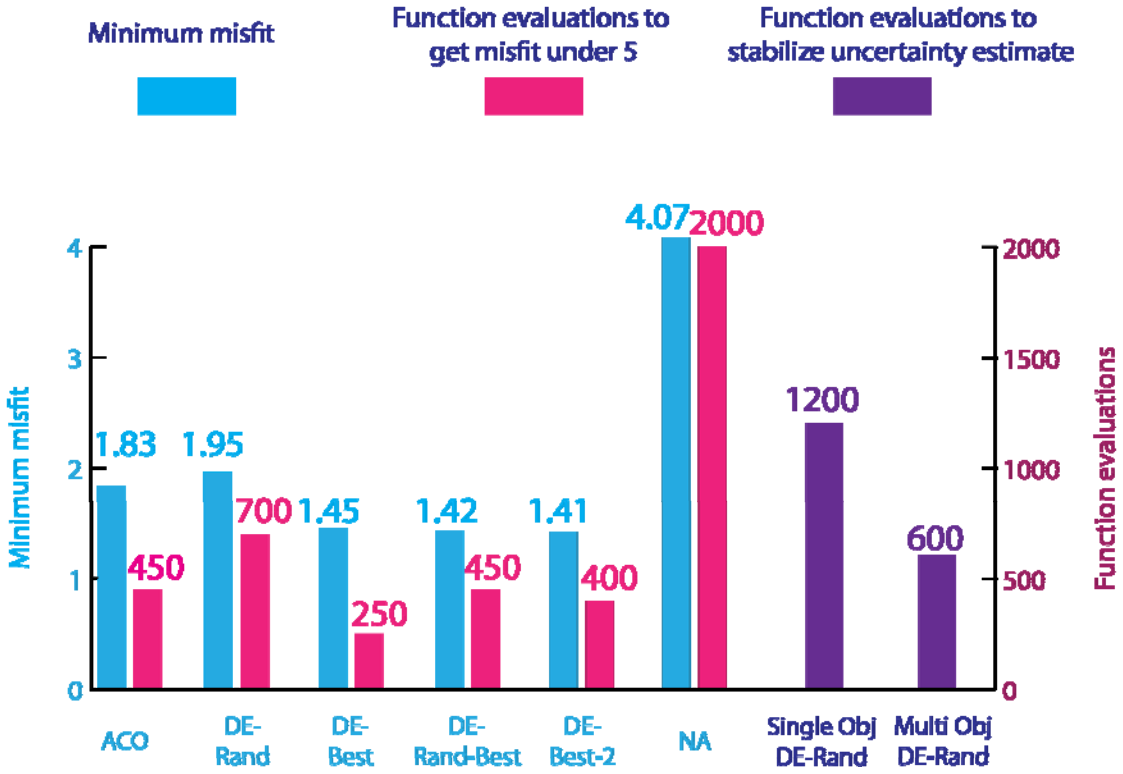

Figure 1: Improvements introduced to history matching and uncertainty quantification framework by bringing ACO_R, DE and multiobjective optimization

In the Teal South reservoir the difference between the performance of ACO_R, DE and NA was marginal due to the use of a relatively simple model with just eight free parameters and the univariate objective function based on the data from a single-well. In high dimensional problems (PUNQ-S3), ACO_R and DE obtain much better models in comparison with NA.

As seen in figure 1, for a relatively complex model, by using ACO_R and DE we see improvements both in minimum misfit obtained in history matching and number of required simulations to achieve this quality of match.

Ant colony optimization and differential evolution offer more flexibility to the end user for tuning the performance of assisted history matching framework. While the neighbourhood algorithm adjusts its sampling using a single parameter, both ACO_R and DE offer two tuning parameters to adjust the exploration/exploitation capabilities of the search. We also demonstrated that ACO_R and DE are able to provide satisfactory results concerning the initial random seed value.

We see that the sampling efficiency of the algorithms can have a direct impact on uncertainty estimates. A wide sampling of the search space and finding multiple good quality models ensures a more reliable prediction of future reservoir performance. Different algorithms were able to find different models of similar history match quality located in different regions of the parameter space.

Assessing the effect of additional data clearly showed the impact of these data on the calculated range of recovery estimates. The additional data resulted in reduced uncertainty bands and better predictions of ultimate oil recovery.

Finally, multiobjective versions of ACO_R and DE were developed in chapter 6 and their performances were evaluated. Based on numerical tests on three benchmark functions, DE-Rand was selected as the algorithm of choice for multiobjective history matching on PUNQ-S3 reservoir model. It is concluded that the multiobjective approach improves both match quality (minimum misfit) and the number of simulations required to obtain a stable uncertainty estimate for ultimate oil recovery.

7.3 Recommendations

Considering the discussions through the different chapters of this thesis, the following ideas can be considered as research topics for future work. The recommendations for future research can be classified into three main groups: further development of codes, visualization of the results and future areas of applications.

7.3.1 Further Code Developments

One of the major areas that can be considered for further research is improving current codes to handle more complex cases and/or better performance of the optimization. There is also a great opportunity to bring more recent developments in optimization algorithms for history matching and uncertainty quantification frameworks. It is recommended to investigate the following issues:

7.3.1.1 Handling Mixed Variables

In a typical history matching problem, we may have a case where discrete (e.g. facies type [Liu, 2005]) and continuous variables coexist in the list of uncertain parameters. Usually two approaches are chosen in these situations. Either the continuous variables are discretized and the whole problem is treated as a discrete optimization or the discrete variables are considered as continuous and after the optimization the results of discrete parameters are rounded to the nearest possible value.

One of main benefits of the ACO_R algorithm introduced in chapter 3 is that it can simultaneously handle both continuous and discrete variables in the optimization process. Socha [2008] shows some examples of this application. He applied mixed variable ACO_R for designing thermal insulation systems, coil springs and pressure vessels. In this thesis, only continuous variables were considered in the history matching process, but some discrete variables could also be incorporated in the proposed framework like facies type, well location, etc. Application of ACO_R for handing mixed variable problems in history matching (and other areas of reservoir/production engineering) looks to be an interesting field of research.

For the differential evolution algorithm, there are new developments to enable this optimization technique to handle mixed variable challenges. Most of these works [Lampinen and Zelinka, 1999, Pampara et al. 2006] are actually real-coded DE algorithms which use encoding and decoding mechanisms to deal with mixed variable problems. Recent work of Datta and Figueira [2010] is a good example of the efforts for proposing a mixed-variable DE.

The challenge of mixed variables is not limited to the sampling of parameter space in history matching. In the uncertainty quantification phase, we face the same problem. The framework used in this thesis is not able to handle mixed-variable problems. This problem arises from interpolation of the probability in continuous space using Voronoi cells. However when there are discrete variables, this procedure is not helpful. For uncertainty quantification in mixed variable environments using Bayesian methods, there have been recently some proposed solutions. Interested readers can refer to Shenoy [2006], Yuan and Druzdzel [2007] and Cobb et al. [2007] for a detailed discussion on the benefits and limitations of each approach. The application of a mixed variable uncertainty quantification technique in petroleum engineering and specially history matching area calls for further research.

7.3.1.2 Dynamic Tuning of Parameters

In our current work, the tuning parameters of ant colony optimization and differential evolution were set at the beginning of the optimization and remained constant during the run. This is the simplest scheme of these algorithms. There are many works on adaptive tuning of evolutionary algorithms

One of the simple proposals for allowing a dynamic change in the behavior of evolutionary algorithms is the change of any tuning parameters at a constant rate during the optimization. Moving to more advanced tuning mechanisms, we have *adaptive* agent-based algorithms. In this class of methods, the tuning parameters of algorithms are modified using the information provided during the optimization. For example, Omran et al. [2005] introduced a self adaptive differential evolution (SDE), where the parameter F was computed separately for each individual member. First, the scaling factor was randomly initialized for all members and then, based on a random choice of

three members, the new *F* value was computed for the selected individual. Other strategies for having adaptive differential evolution algorithms are discussed in Zhang et al. [2009].

Gong et al. [2009 b] proposed a clustering method for adaptive parameter control of continuous ant colony optimization (COAC). In this work, state of optimization is evaluated at different stages using a clustering analysis and tuning parameters are dynamically adjusted according to the optimizations state. They conclude that adaptive tuning results in increased convergence speed and solution accuracy.

Madadgar and Afshar [2009] propose an adaptive version of the ACO_R algorithm, where the search localization parameter (*q*) was dynamically changed during optimization. In the first iterations, a relatively large value is selected for *q* and at later stages; *q* is dynamically decreased according to the mean value of objective functions at each iteration. This helps to have a global exploration of search space in the early stages of optimization and an exploitative behavior in final iterations to refine the promising regions.

As discussed in many works, dynamic tuning of ant colony optimization and differential evolution give more powerful algorithms for tackling difficult optimizations tasks and should be incorporated into current algorithm in the thesis for future work.

7.3.1.3 Other Agent-Based Optimization Methods

Agent-based evolutionary optimization is an umbrella term covering many algorithms. Further to ant colony optimization and differential evolution used in this study, we can also review other options in the literature which have not yet been applied (or have only limited applications) for reservoir engineering and history matching problems. In this context, the following algorithms are recognized to have a potential application in history matching field.

Artificial immune systems are one of the fastest growing areas of research in biologically-inspired computing science [Hart et al. 2010]. The idea behind this algorithm is the cooperative work of the natural immune systems to eliminate invading

pathogens. Immune systems are capable of recognition, selecting, leaning and adoption [Timms and Neal, 2000]. In recent years many people have tried to tackle difficult engineering problems using artificial immune systems [Lee et al. 2009].

The Intelligent Water Drop (IWD) algorithm is another novel development in agent-based optimization [Shah Hosseini, 2007]. Rivers often find optimal paths in their ways from the source to their destination. The IWD algorithms benefits from action and reactions between water drops and riverbeds. Several intelligent water drops cooperate in this algorithm to change the environment in such a way that an optimal path is selected.

The bee colony optimization algorithm is also one of the latest developments in the swarm intelligence paradigm [Karaboga, 2005]. This technique is based on the behavior of a honey bee colony in finding flower patches. The bee colony algorithm shares some similarities with ant colony optimization, although some concepts such as waggle dance is only defined in the bee colony algorithm. Recently, a multiobjective version of this algorithm has been applied to optimize electro-chemical machining parameters [Pawar et al. 2008].

Quantum Inspired Evolutionary Algorithms (QIEAs) [Zhang, 2010] are also a new and promising field of research in evolutionary computation. These algorithms use computational methods based on principles of quantum mechanics, such as qubits, superposition and quantum gates to solve various optimization problems in a probabilistic framework.

Estimation of distribution algorithms (EDA) are a branch of evolutionary algorithms where the statistical distribution of the search is the central concept, as opposed to populations and individuals. The probabilistic model obtained from these distributions has a significant influence on the performance of the EDA. Generally three classes of EDA can be found in the literature. Based on the ability to capture dependencies between variables, EDA may not consider any dependency (for example Population-Based Incremental Learning (PBIL), compact Genetic Algorithm (cGA)), or may only consider dependencies between pairs of variables (e.g. Mutual Information Maximization for Input Clustering (MIMIC) and Bivariate Marginal Distribution

Algorithm (BMDA), or can work with multiple dependencies (e.g. Estimation of Bayesian Networks Algorithm (EBNAs), Estimation of Gaussian Networks Algorithms (EGNAs) and different flavors of the Bayesian Optimization Algorithm (BOA)).

EDAs have started to play an important role for real-world problems. Stewart et al. [2008] demonstrated the application of regularity modeling multiobjective estimation of distribution algorithm (RM-MEDA) for optimization of 3D turbine blades. Oschoa [2010] discusses shrinkage EDAs for solving computationally-expensive optimization challenges. Mendiburu et al. [2010] provides an overview of different strategies for parallelization of EDAs, which seems necessary for real-world engineering problems. Recently Abdollahzadeh et al. [2011] introduced the application of EDAs for history matching problem.

The research into these novel algorithms and especially their application to reservoir engineering problems is in the early stages and expect a wider acceptance in future.

7.3.1.4 Hybrid Algorithms

Population-based evolutionary optimization algorithms are powerful tools for solving difficult engineering problems, including history matching. However, in certain cases, a specific algorithm may perform better than other available solvers. Researchers in the field of computer science have devoted significant amount of research to propose hybrid versions of stochastic population algorithms, mainly focusing on two directions; firstly to improve the performance of algorithms (convergence speed, exploration/exploitation, etc) and secondly, to reduce the number of control parameters, so making then easier to use in practical applications.

There are numerous publications where hybrid versions of population-based stochastic optimization algorithms have been introduced and applied to various problems. Ali et al. [2009] introduced a hybrid version of differential evolution and ant colony optimization, ACDE, and applied it to water resource problems. In this framework, a continuous ant colony algorithm was used to refine the results of differential evolution at each iteration. It was claimed that this approach significantly increases the convergence speed of differential evolution.

Differential evolution particle swarm optimization (bare bones differential evolution) which is a hybrid of barebones PSO and DE [Omran et al. 2009], does not use classical PSO parameters like inertia weight, acceleration parameters and removes DE scaling parameter. The only parameter to tune is the probability of recombination.

Another example is the hybridization of continuous ant colony system (CACS) and Tabu Search (TS) for minimization of multi-minima functions [Karimi et al. 2010]. They compared the results with several ant colony based methods (including ACO_R) and conclude that the proposed approach can increase convergence speed of ant colony optimization, especially in low dimensional problems. Recently, some new ideas in hybridization of ant colony optimization and artificial immune systems were presented in Gao [2007].

Most hybrid versions of algorithms reported in the literature, have shown an improvement in computational properties of algorithms in comparison with individual original contributions. The application of hybrid methods for history matching should be considered as a brilliant opportunity for future studies.

7.3.1.5 Parallelization of the Codes

Parallel and distributed computing can be used to speedup the evolutionary algorithms. The rapid evolution of technology and constant decrease in price of hardware make parallelization very popular in recent years. Different architectural criteria must be considered in designing a parallel optimization code. The efficiency of the implemented code depends on shared/distributed memory, homogenous/heterogeneous hardware, local/large network and many more factors. Depending on available resources, several architectures have been proposed in the literature. Usually, these techniques are recognized by the way they manage communication between processors and the way optimization agents are distributed between them.

Parallelization of evolutionary algorithms can be done in direct or island-based [Mendiburu et al. 2010]. In direct transformation of a serial code to parallel version, no change is made in main steps of evolutionary algorithm. In island-based methods,

several sub-populations are created using different topologies and these island exchange information at certain points during optimization. On the other hand, implementations can be synchronous or asynchronous [Sharma, 2009]. In synchronous operations, we should wait for results of all processors to finish. This may decrease the efficiency of parallelization, because some simulations are input-dependent and take longer to complete. This problem can be solved with an asynchronous implementations, where the next generation of solutions can be produced using some of the completed runs in previous iteration.

Many authors have proposed parallel versions of ant colony optimization and differential evolution. Stutzle [1998] discusses some approaches to parallelize discrete ant colony optimization (ACO) algorithms. Apparently, only Lin et al. [2007] have proposed a parallel continuous ACO algorithm, based on the scheme introduced by Zhang et al. [2006]. Zhu et al. [2010] introduced a parallel differential evolution using the emerging desktop-based graphics processing unit (GPU) technology. Furthermore, several works have proposed parallel multiobjective agent-based optimization algorithms. Montano et al. [2010] introduced a parallel differential evolution. Talbi et al. [2008] review other strategies for parallelization of multiobjective evolutionary optimization algorithms.

Among different optimization algorithms applied for history matching in this thesis, only the Neighbourhood algorithm (NA) has the capability to run in a parallel mode (on different nodes of a cluster). Ant colony and differential evolution codes are currently programmed to run in serial mode, which means the runs can only be performed on a single node of the cluster. In order to fully benefit from these techniques, it is recommended to add this feature to existing codes in R language. Several packages provide the communication platform required for parallel computing in R language such as Rmpi [Yu, 2010]. The SNOW (Simple Network of Workstations) package [Tierney et al. 2008] works with PVM and MPI standards to hide the background communications. Recently REvolution Computing [2009] has introduced the *foreach* package which provides a tool to run for loops in parallel. Since in current codes, each generation of populations is evaluated using a *for loop*, this package seems a straightforward way to parallelize current ant colony and differential evolution codes.

7.3.1.6 Fuzzy Rule-Based Systems

All of the methods discussed so far in the history matching research community need some higher level supervision by a human. This is required to check the results, select appropriate ones and eliminate the unrealistic models or the ones that will not fit with the scope of project. In this context, we are discussing an assisted history matching framework. A very interesting area is the coupling of history matching algorithms with rule-based expert systems to replace the human decision maker. This can gradually transform current assisted history matching frameworks to fully automatic history matching frameworks.

Rule-based systems are a form of artificial intelligence and tend to embody the knowledge of human experts in a computer program. Fuzzy set theory [Zadeh 1965] provides a convenient and powerful way to model vague data according to the subjective imagination of a decision maker. Using fuzzy set theory, we can have a collection of fuzzy rules to evaluate relationship of input parameters and system response with expert knowledge being embedded in the system [Iqbal and Dar, 2009].

Fuzzy rule-based systems can bring reservoir engineering sense to history matching frameworks. The essence of this system is a set of IF-THEN rules derived by engineering knowledge. As shown in figure 2, fuzzy rule-based system can check the results of optimization against expert knowledge and provide a guideline to optimization algorithm for producing next set of solutions.

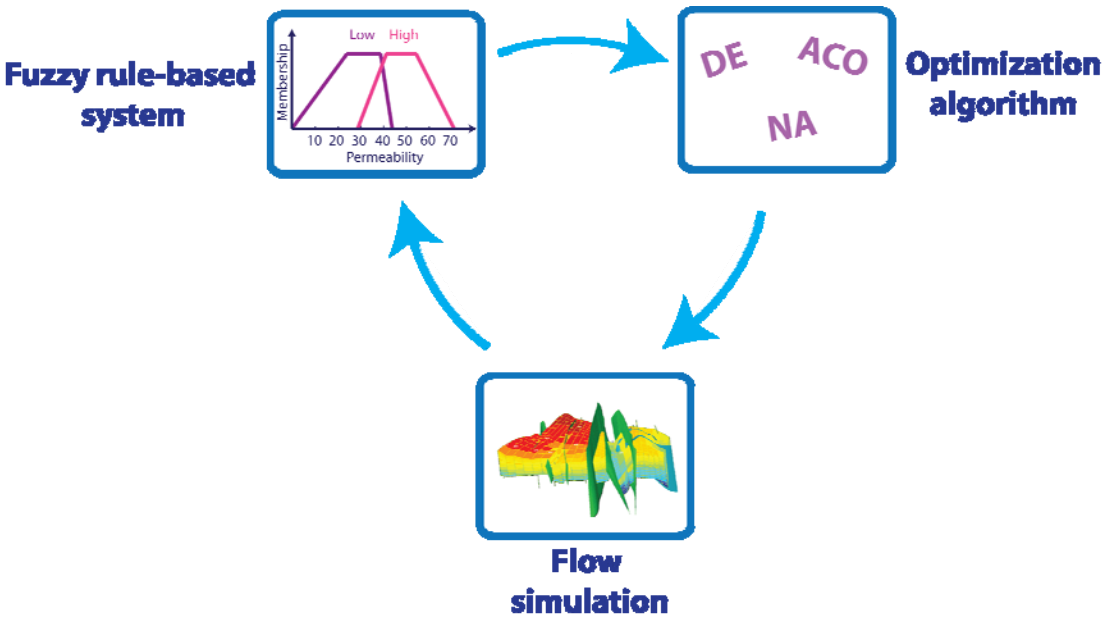

Figure 2: Suggested workflow for bringing reservoir engineering knowledge to history matching by integrating a fuzzy rule-based system

Fuzzy set theory can also be used to define a fuzzy objective function. Rommelfanger [2007] provides a critical survey of different methods in optimization of fuzzy objective functions. The application of fuzzy functions for multiobjective optimization of electromagnetic devices has been demonstrated by Chiampi [et al. 1998]. The integration of fuzzy logic into history matching framework, both as a rule-based system and objective function definition, are very promising research topics for future.

7.3.1.7 Reducing the Number of Function Evaluations in Optimization

Saving computational costs in evolutionary optimization by reducing the number of objective function evaluations has always been an interesting field of research. Jin [2005] presented a taxonomy of approaches for incorporating knowledge into evolutionary algorithms. He identified functional approximation and evolutionary approximation as main tools in reducing the computational time of evolutionary optimization. Examples of functional approximation or meta-models include response surface methods, Gaussian processes (kriging), radial basis functions, neural networks and support vector machines. The evolutionary approximation tries to reduce function

evaluations by estimating an individual's misfit from other similar individuals in the population. Two widely-used examples in this category are fitness inheritance and clustering.

In our study, for both single and multi objective optimizations, we directly used objective function values obtained from simulator. For future studies, there is scope to combine current optimization techniques with advanced knowledge incorporation schemes to enhance their performance and reduce computational costs.

7.3.1.8 Reducing the Number of Objectives in Optimization

It has been observed that the performance of many evolutionary multiobjective optimization methods declines when the number of objectives increases. In these situations, it is recommended that dimensionality reduction schemes for multiobjective optimization [Brockhoff and Zitzler, 2006] and indicator based algorithms [Ulrich et al. 2010] be used. Also when the number of objectives increases in a problem, non-dominated solutions based on the concept of Pareto-dominance become more seldom and the search decelerates. For tacking this problem, some other measures of dominance may be employed in these problems, for example ranking dominance procedure [Kukkonen and Lampinen, 2007].

In chapter 6, we applied a bi-objective differential evolution algorithm to history match the PUNQ-S3 problem. For this purpose, we grouped different match criteria to formulate a bi-objective problem. In dealing with more variables, it is recommended to apply schemes mentioned above to enhance the performance of multiobjective history matching.

7.3.1.9 Other Ideas for Further Development of Codes

Recently contour methods have been proposed to explore the landscape of the fitness function [Lin et al. 2008]. This method, which is inspired by the contours in topography, can be embedded in to any population-based algorithm to improve its performance. Lin et al. [2008] claimed a significant improvement in terms of the solution quality and convergence of the search when they used this method in a simple

genetic algorithm. Certainly this approach looks very promising to be used together with proposed population-based algorithm used in this thesis for history matching.

Another interesting idea proposed recently by Salhi and Toreyen [2010] is a game theory based multi-agent (algorithm) system for expensive real-world optimization problems. Within this framework, different algorithms (genetic algorithm and simulated annealing in this work) act as agents in the problem and play a cooperative-competitive game to decide which algorithm, if any, should be dropped from the list of solvers. They suggest that, by choosing the best suited algorithm(s) for individual problems, the best use of both algorithms and computational resources is guaranteed. As history matching is also considered as a computationally-expensive problem, the use of such "smart" collection of solvers brings additional benefits in comparison with current single algorithm frameworks.

Hernandez-Diaz et al. [2008] proposed to seed the evolutionary multiobjective optimization algorithm using gradient-based information. Unlike many works which combine these two types of algorithms during the optimization, in their work, only initial stage (seeding) of evolutionary optimization (NSGA-II) is performed using a gradient-based scheme (steepest descent). They conclude that, their proposed hybrid method results in a significant decrease in the number of function evaluations, while maintaining optimal solution quality. This idea can be incorporated into our proposed methodologies to increase their computational efficiency.

7.3.2 Visualization of Results

In this thesis, many visualization techniques have been used to report the results, such as sampling history figures, boxplots, heatmaps, etc. Further to these relatively basic graphs, it is recommended to use advanced visualization techniques in future developments. There are two separate areas in this context

7.3.2.1 Visualization of Optimization Algorithms

Recent developments in visualization of optimization algorithms open windows of opportunities to gain better insights into the performance of agent-based techniques. One of the most powerful visualization toolboxes for optimization algorithms is

VISPLORE, developed by Khemka and Jacob [2010]. Originally designed for visualization of particle swarms, VISPLORE provides an interactive environment for analysis of agent-based optimization results. Figure 3 provides some schematic example graphs produced by this toolbox.

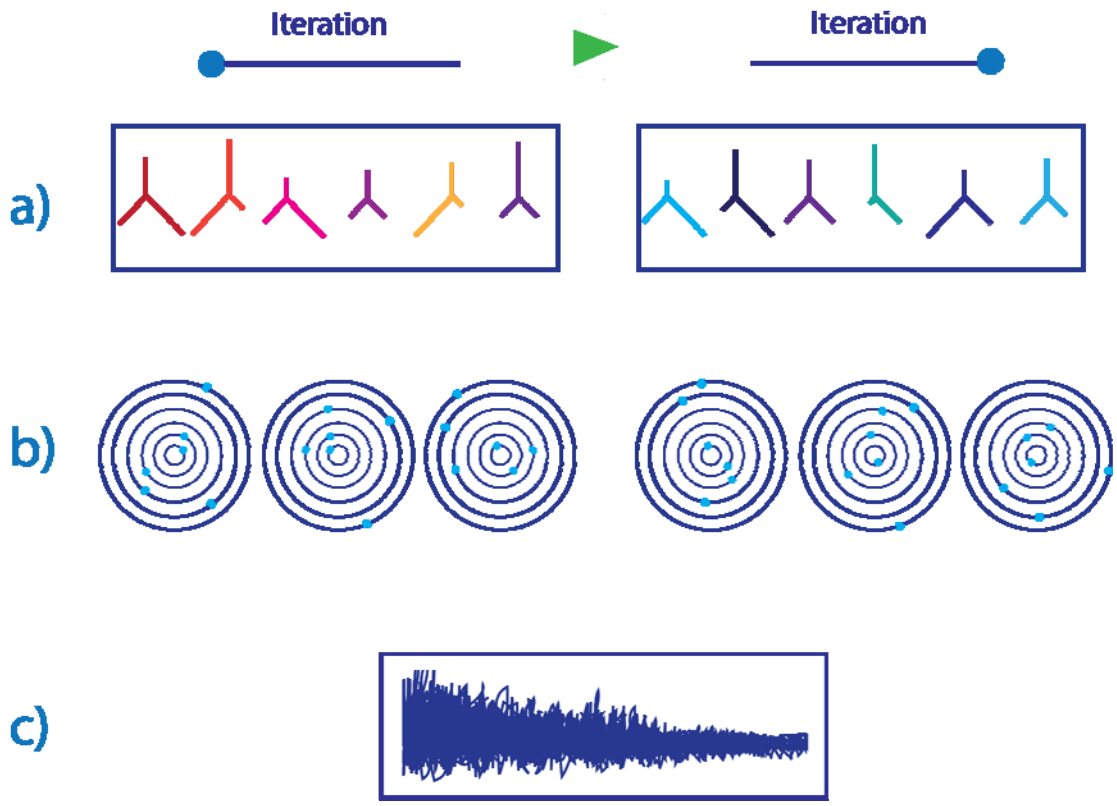

Figure 3: Schematic graphs produced by VISPLORE

Figure 3 (a) shows the star plots where a N-Dimensional vector is projected onto the 2-D space using a set of D lines originating at the origin and equally spaced over 360°. The length of each line represents the value of the corresponding dimension. Figure 3 (a) presents star plots for an evolutionary algorithm in optimization of a three parameter problem with six individuals in each generation (iteration). Concentric circle plots in figure 3 (b) provide a dimensional view of the population in agent-based optimizations. Each set of circles represent one dimension of problem. Each circle of the set shows one agent, with the inner most circle being agent 1. The points on circles show parameter values. Another innovative plot in VISPLORE package is the sound plot. This plot

utilizes a sound wave to indicate the convergence of optimization. The sum of fitness values is converted to a sine wave according to following equation:

$$\sum_{i=1}^{\mu} \sin[(1440 + 40 f(p_i) * 2\pi)]$$

This sound wave then can be played and by listening to it, one can hear a noise at initial stages and smooth wave at final steps of optimization. Other graphs in this toolbox are parallel coordinates, density plots, convergence plots.

We have discussed multiobjective optimization problem for two objectives in chapter 6. For higher number of objectives, the Pareto front will form a space and in this case we should use advanced visualization techniques to aid the decision maker in selecting results. For this purpose we can refer to Blasco et al. [2008].

7.3.2.2 Visualization of Simulations

In order to compare history matched models with true reservoir parameters or compare two different history matched models, the most common approach in reservoir engineering community is to plot two different models in two windows and have a visualize comparison of reservoir sections in these models. As a step change, recently advanced visualization toolkits have come into play. For example, Kongsberg has developed the SIM Reservoir product to visualize different reservoir models. Figure 4 shows a snapshot of this software [SIM Reservoir, 2010]. The major benefit of this new toolkit is that multiple models/data can be visualized in a single window, which transforms the traditional comparison of observed vs. simulated models.

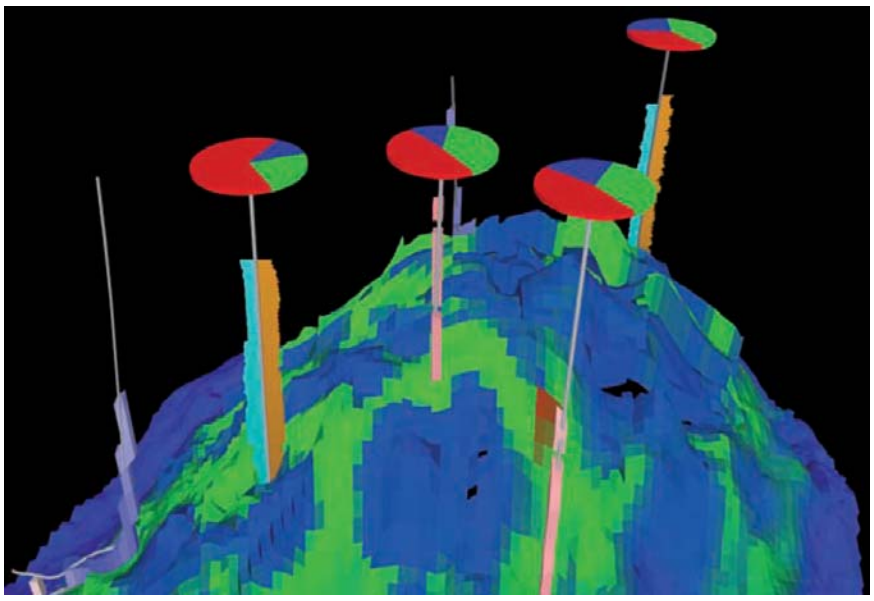

Figure 4: Comparison of observed and simulated data at wells shown as histograms at well locations; pie charts show production data. Color of the cell can show the level of match between two different reservoir models

7.3.3 Further Application Areas

In chapter 1, we introduced the customers of this thesis as reservoir management teams to update their reservoir models. The techniques and workflows introduced in this thesis are not restricted to history matching and can be applied in many other areas. In following paragraphs we introduce some of these fields.

One of the immediate areas where novel optimization techniques can be directly applied is the seismic history matching [Gosselin et al. 2003] where a set of time lapse seismic (4D) data is also considered as a compliment to production to condition the reservoir model. It is known that [Williams, 2010] using both seismic and production data for history matching can be challenging in large-scale industrial problems. Application of multiobjective optimization for joint handling of production and seismic in particular seems very interesting for this problem.

In exploration and production business, there are many other diverse areas where proposed optimization algorithms can be applied. For example van Essen et al. [2009] considered short term and long term life cycle optimization of fields considering Net

Present Value (NPV). Determining optimum well locations is also one of the crucial steps in field development. Many algorithms have been used for well location optimization such as genetic algorithm [Morales et al. 2010]. Other areas in production optimization include gas lift optimization [Djikpesse et al. 2010], pig lift optimization* [Zhou et al. 2008], prediction of minimum miscibility pressure (MMP) in EOR processes [Emera and Sarma, 2005], wellbore design (optimization of length and positioning of multiple tubular strings) [Kumar et al. 2010], pressure optimization in separators in production plants [Kylling, 2009] and many more. Application of novel optimization methods in above fields is a very interesting arena for future research.

7.4 And Finally We Are Going To …

IBM is going to have machines with processing powers up to 100 petaFLOPS in 3 years. FLOPS (or flops or flop/s) is an acronym meaning Floating point Operations Per Second. The FLOPS is a measure of a computers performance, especially in fields of scientific calculations. It is believed that human brain has a power of 10 petaflops. (number of neurons times average number of connections between neurons times neuron firing capacity per second). Imagine this computing power is coupled with reservoir simulators like GIGAPOWERS where geological-scale model is used as the flow simulation model. With this coupling, performing tens of thousands of simulation within a realistic timeframe will be possible. Considering this position, the author believes that, in future, bringing more engineering knowledge in black-box optimization workflows should be considered as a very important task. Since the ultimate goal of reservoir engineering study, including history matching and uncertainty quantification, is the improved decision making, working on improvement of current decision making frameworks must be on agenda.

* Pig lift is new artificial lift technology for reducing liquid accumulation and increasing two-phase flow stability

My guide and I came on that hidden road
to make our way back into the bright world;
and with no care for any rest,

We climbed, he first, and I following, until I saw,
through a round opening, some of those things
of beauty heaven hearts. It was from there
that we emerged, to see, once more, the stars.

Dante, Inferno, XXXIV, 133-139

References

Abbaspour, K., Schulin, R., Genuchten, M [2001] Estimating Unsaturated Soil Hydraulic Parameters Using Ant Colony Optimization, Advances in Water Resources, 24, 827-841

Abbass, H. A [2001] A Memetic Pareto Evolutionary Approach to Artificial Neural Networks, In Stumptner, M., Corbett, D.R., Brooks, M (Eds.) Canadian AI 2001, Springer LNCS, volume 2256, 1-12

Abbass, H.A., Sarker, R [2002] The Pareto Differential Evolution Algorithm, International Journal on Artificial Intelligence Tools, volume 11, issue 4, 531-552

Abbass, H.A [2002] The Self-Adaptive Pareto Differential Evolution Algorithm, In Congress on Evolutionary Computation (CEC 2002), Piscataway, New Jersey, 831-836

Abdollahzadeh, A., Reynolds, A., Christie, M., Corne, D., Williams, G., Davies, B.J [2011] Estimation of Distribution Algorithms Applied to History Matching, SPE 141161, Reservoir Simulation Symposium, The Woodlands, Texas, USA, 21-23 February

Abu El Ela, M [2007] Data Analysis Methodology for Reservoir Management, SPE 106899, EUROPEC/EAGE Annual Conference and Exhibition, London, UK, 11-14 June

Agostinetti, N., Spada, G., Cianetti, S [2004] Mantle Viscosity Inference: A Comparison between Simulated Annealing and Neighbourhood Algorithm Inversion Methods, Geophysics Journal, 157, 890-900

AIAA [1998] Guide for the Verification and Validation of Computational Fluid Dynamics, G-077-1998-e, American Institute of Aeronautics and Astronautics

Alam, S., Bui, L, Abbas, H, Barlow, M [2006] Pareto Meta-Heuristics for Generating Safe Flight Trajectories Under Weather Hazards, Lecture Notes in Computer Science, 4247, 829-836

Al-Ani, A [2005] Feature Subset Selection Using Ant Colony Optimization, International Journal of Computational Intelligence, volume 2, number 1, 53-58

Alaya, I., Sonon, C., Ghedira, K [2007] Ant Colony Optimization for Multi-Objective Optimization Problems, 19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI), Patras, Greece, 29-31 October

Al-Dhafeeri A, Nasr-El-Din H [2007] Characteristics of High Permeability Zones Using Core Analysis and Production Logging Data, Journal of Petroleum Science and Engineering, 55, 18-36

Alfaro, M.E., Zoller, S., Lutzoni, F [2003] Bayes or Bootstrap? A Simulation Study Comparing the Performance of Bayesian Markov Chain Monte Carlo Sampling and Bootstrapping in Assessing Phylogentic Confidence, Molecular Biology and Evolution, volume 20, issue 2, 255-266

Ali, M., Pant, M., Abraham, A [2009] A Hybrid Ant Colony Differential Evolution and its Application to Water Resources Problems, IEEE conference on Nature & Biologically Inspired Computing, NaBIC, 1133-1138

Alpern, B., Carter, L [1991] The Hyperbox, Proceedings of IEEE Visualization Conference, San Diego, CA, USA, 22-25 October

Alvarez-Benitez, J., Everson, R., Fieldsend, J [2005] A MOPSO Algorithm Based Exclusively on Pareto Dominance Concept, Evolutionary Multiobjective Optimization 2005, LNCS 3410, Springer, 459-473

Anchliya A [2006] A Review of Seismic While Drilling Techniques: A Journey from 1986 to 2005, SPE 100352, EUROPEC/EAGE Annual Conference and Exhibition, Vienna, Austria, 12-15 June

Anderson, L [2009] An Ensemble Adjustment Kalman Filter for Data Assimilation, Mon Wea Rev, 129, 2884-2903

Angira, R., Babu, B.V [2006] Performance of Modified Differential Evolution for Optimal Design of Complex and Non-linear Chemical Processes, Journal of Experimental and Theoretical Artificial Intelligence, volume 18, issue 4, 501-512

Angus, D [2007] Population-Based Ant Colony Optimization for Multi-Objective Function Optimization, In Randall, M., Abbass, H.A., Wiles, J (Eds.) ACAL 2007, LNAI 4828, 232-244, Springer-Verlag, Berlin Heidelberg

Angus, D., Woodward, C [2009] Multiple Objective Ant Colony Optimization, Swarm Intelligence, 3, 69-85

Anterion, F., Eymard, R., Karcher, B [1989] Use of Parameter Gradients for Reservoir History Matching, SPE 18433, Symposium on Reservoir Simulation, Houston, Texas, USA, February 6-8

Aprilia A, Li Z, McVay D, Lee W [2006] Quantifying Uncertainty in Original-Gas-in-Place Estimates with Bayesian Integration of Volumetric and Material Balance Analysis, SPE 100575, Gas Technology Symposium, Calgary, Canada, 15-17 May

Aristeguieta, O [2008] Multiobjective Portfolio Optimization of Upstream Petroleum Projects, University of Adelaide, M Eng Thesis

Arwini, S., Stephen, K.D [2010] A New Method to Improve Convergence Rates with Seismic History Matching, SPE 131545, EUROPEC/EAGE Annual Conference and Exhibition, Barcelona, Spain, 14-17 June

Azi, C., Gbo, A., Whittle, T., Gringarten, A [2008] Evaluation of Confidence Intervals in Well Test Interpretation Results, SPE 113888, EUROPEC Conference and Exhibition, Rome, Italy, 9-12 June

Azzam, M., Mousa, A [2009] Using Genetic Algorithm and TOPSIS Technique for Multiobjective Reactive Power Compensation, Electric Power Systems Research, Article in Press

Babu, B.V., Jehan, M.M.L [2003] Differential Evolution for Multiobjective Optimization, In Proceedings of the 2003 Congress on Evolutionary Computation (CEC 2003), Canberra, Australia, volume 4, 2696-2703

Babu, B.V., Chakole, P., Syed Mubeen, J.H [2005] Multiobjective Differential Evolution (MODE) for Optimization of Adiabatic Styrene Reactor, Chemical Engineering Science, 60, 4822-4837

Babu, B.V., Gujarathi, A [2007] Multi-Objective Differential Evolution (MODE) for Optimization of Supply Chain Planning and Management, Proceedings of IEEE Congress on Evolutionary Computation (CEC 2007), Singapore, 2732-2739, September 25-28

Babu, B.V., Gujarathi, A., Katla, P., Laxmi, V.B [2007] Strategies of Multi-Objective Differential Evolution (MODE) for Optimization of Adiabatic Styrene Reactor, Proceedings of International Conference on Emerging Mechanical Technology-Macro to Nano (EMTMN-2007), BITS Pilani, February 16-18, 243-250

Babu, B.V., Munawar, S.A [2007] Differential Evolution Strategies for Optimal Design of Shell-and-Tube Heat Exchangers, Chemical Engineering Science, volume 62, issue 14, 3720-3739

Bachman, R.C., Harding, T.G., Settari, T., Walters, D [2003] Coupled Simulation of Reservoir Flow, Geomechanics, and Formation Plugging With Application to High-Rate Produced Water Reinjection, SPE 79695, Reservoir Simulation Symposium, Houston, Texas, 3-5 February

Ballester, P., Carter, J [2007] A Parallel Real-Coded Genetic Algorithm for History Matching and Its Application to a Real Petroleum Reservoir, Journal of Petroleum Science & Engineering, volume 59, 157-168

Banks, A., Vincent, J., Wudi, C [2007] A Review of Particle Swarm Optimization, Part 1: Background and Development, Natural Computing, 6, 467-484

Barker, J.W., Cuypers, M., Holden, L [2001] Quantifying Uncertainty in Production Forecasts: Another Look at the PUNQ-S3 Problem, SPE 74707, SPE Journal, volume 6, Number 4, 433-441

Baskar, N., Saravanan, R, Asokan, P., Prabhaharan, G [2004] Ants Colony Algorithm Approach for Multiobjective Optimization of Surface Grinding Operations, Int J Adv Manuf Technol, 23, 311-317

Begg, S.H., Smit, N [2007] Sensitivity of Project Economics to Uncertainty in Type and Parameters of the Price Models, SPE 110812, Annual Technical Conference and Exhibition, Anaheim, California, 11-14 November

Beghein, C., Resovsky, J., Trampert, J [2002] P and S Tomography Using Normal-Mode and Surface Waves Data with a Neighbourhood Algorithm, Geophysics journal, 149, 646-658

Beghein, C., Trampert, J [2004] Probability Density Functions for Radial Anisotropy From Fundamental Mode Surface Wave Data and the Neighbourhood Algorithm, Geophysics Journal, 157, 1163-1174

Behrenbruch, P., Turner, G., Backhouse, A [1985] Probabilistic Hydrocarbon Reserves Estimation: A Novel Monte Carlo Approach, SPE 13982, Offshore Europe Conference, Aberdeen, September 10-13

Benayoun, R., de Montgolfier, J., Tergny, J., Laritchev, O [1971] Programming with Multiple Objective Functions, Step Method (STEM), Mathematical Programming, 1, 366-375

Best, L., den Berg, F [2006] Smart Fields - Making the Most of Our Assets, SPE 103575, Russian Oil and Gas Technical Conference and Exhibition, Moscow, Russia, 3-6 October

Beven, K., Binley, A [1992] The Future Distributed Models – Model Calibration and Uncertainty Prediction, Hydrological Processes, 6, 279-298

Bhaskar, V., Gupta, K, Ray, K [2000] Applications of Multiobjective Optimization in Chemical Engineering, Review Chem Eng, 16, 1-54

Bianco, A., Cominelli, A., Dovera, L., Naevdal, G., Valles, B [2007] History Matching and Production Forecast Uncertainty by Means of the Ensemble Kalman Filter, A Real Field Application, SPE 107161, SPE/EUROPEC Conference and Exhibition, London, UK, 11-14 June

Biesbroek, R [2006] A Comparison of the Differential Evolution Method with Genetic Algorithms for Orbit Optimization, in 57th International Astronautical Congress, Valencia, Spain

Bilchev, G., Parmee, C [1995] The Ant Colony Metaphor for Searching Continuous Design Spaces, In Proceedings of AISB Workshop on Evolutionary Computing Lecture Notes in Computer Science 993, Berlin: Springer-Verlag

Birchenko, V., Demyanov, V., Konopczynski, M,. Davies, D [2008] Impact of Reservoir Uncertainty on Selection of Advanced Completion Type, SPE 115744, Annual Technical Conference and Exhibition, Denver, Colorado, USA, 21-24 September

Blasco, X., Herrero, J.M., Sanchis, J., Martinez, M [2008] A New Graphical Visualization of N-Dimensional Pareto Front for Decision-Making in Multiobjective Optimization, Information Sciences, 178, 3908-3924

Blasone, R.S., Vrugt, J.A., Madsen, H., Rosbjerg, D., Robinson, B.A., Zyvoloski, G.A [2008] Generalized Likelihood Uncertainty Estimation (GLUE) Using Adaptive Markov Chain Monte Carlo Sampling, Advances in Water Resources, 31, 630-648

Blockley, D., Godfrey, P [2000] Doing it Differently: Systems of Rethinking Construction, Thomas Telford, London, 2000

Blum, C [2005] Ant Colony Optimization: Introduction and Recent Trends, Physics of Life Reviews, 2, 353-373

Blum, C., Socha, K [2005] Training Feed-Forward Neural Networks with Ant Colony Optimization: An Application to Pattern Classification, Fifth International Conference on Hybrid Intelligent Systems, Rio de Janeiro, Brazil, November 6-9

Bonet-Cunha, L., Oliver, D.S., Redner, R.A, Reynolds, A.C [1998] A Hybrid Markov Chain Monte Carlo Method for Generating Permeability Fields Conditioned to Multiwell Pressure Data and Prior Information, SPE 50991, SPE Journal, volume 3, issue 3, 261-271

Boomer, K., Brazier, R [2009] Stochastic Modeling of the Velocity Structure: Beyond Joint Inversion Methods, 11th SAGA Biannual Technical Meeting and Exhibition, Switzerland, 16-18 September

Boss, C. [1999] Production forecasting with UNcertainty Quantification, Netherlands Institute of Applied Geoscience TNO

Bosman, P., Thierens, D [2000] Continuous Iterated Density Estimation Evolutionary Algorithms within the IDEA Framework, In: Pelikan, M., Muhlenbein, H., Rodriguez, A. (Eds.) Proceedings of OBUPM Workshop at GECCO, Morgan-Kaufmann Publishers, San Francisco, CA, 197-200

Bourgault, G [2008] Reducing the Number of History Matching Parameters by Decomposing Static Properties into a Trend and a Residual Component, in VIII International Geostatic Congress, Santiago, Chile, 1091-1097

Branchs, R., Klie, H., Rodriguez, A., Thomas, S., Wheeler, M [2006] A Learning Computational Engine for History Matching, 10th European Conference on the Mathematics of Oil Recovery, Amsterdam, The Netherlands, 4-7 September

Bratvold, R.B., Begg, S.H [2006] Education for the Real World: Equipping Petroleum Engineering to Manage Uncertainty, SPE 103339, Annual Technical Conference and Exhibition, San Antonio, Texas, USA, 24-27 September

Brest, J., Greiner, S., Boskovic, B., Mernik, M., Zumer, V [2006] Self-Adapting Control Parameters in Differential Evolution: A Comparative Study on Numerical Benchmark Problems, IEEE Trans Evolutionary Computation

Brin, S [1995] Near Neighbour Search in Large Metric Spaces, Proceedings of the 21st International Conference on Very Large Data Bases, Zurich, Switzerland, September 11-15

Brizuela, C.A., Gutierrez, E [2005] Multiobjective Go with the Winners Algorithm: A Preliminary Study, Evolutionary Multiobjective Optimization Conference, LNCS 3410, Springer, 206-220

Brockhoff, D., Zitzler, E [2006] Dimensionality Reduction in Multiobjective Optimization: The Minimum Objective Subset Problem, Proceedings of the International Conference of the German Operations Research Society (GOR), 423-429, Springer

Burton, J. L., Franks, N.R [1985] The Foraging Ecology of the Army Ant Eciton rapax: An Ergonomic Enigma?, Ecol. Entomol, 10, 131-141

Bush, M.D., Carter, J.N [1996] Application of a Modified Genetic Algorithm to Parameter Estimation in Petroleum Industry, Intelligent Engineering Systems through Artificial Neural Networks, volume 6, ASME Press, New York

Caers, J [2003] History Matching under a Training Image-based Geological Model Constraints, SPE 74716, SPE Journal, volume 8, number 3, 218-226

Caldwell, R., Heather, D [2001] Characterizing Uncertainty in Oil and Gas Evaluations, SPE 68592, Hydrocarbon Economics and Evaluation Symposium, Dallas, Texas, USA, 2-3 April

Cao, Y. and Wu, Q [1997] Mechanical Design Optimization by Mixed-Variable Evolutionary Programming, Proceedings of the Fourth IEEE Conference on Evolutionary Computation, IEEE Press, 443-446

Capen, E.C [1976] The Difficulty of Assessing Uncertainty, SPE 5579, Journal of Petroleum Technology, volume 28, number 8, 843-850

Cardoso, M.A [2009] Reduced-Order Models for Reservoir Simulation, SPE 129636, International Student Paper Contest, Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 4-7 October

Carlson, M [2003] Practical Reservoir Simulation, ISBN: 0-87814-803-5, PennWell

Carter, J., Ballester, P., Tavassoli, Z., King, P [2006] Our Calibrated Model has Poor Predictive Value: An Example From the Petroleum Industry, Reliability Engineering and System Safety, volume 91, number 10-11, 1373-1381

Castelletti, A., Pianosi, F., Soncini-Sessa, R [2008] Water Reservoir Control Under Economic, Social and Environmental Constraints, Automatica, 44, 1595-1607

Castellini, A., Yeten, B., Singh, U., Vahedi, A., Sawiris, R [2006] History Matching and Uncertainty Quantification Assisted by Global Optimization Techniques, paper A011,

10th European Conference on the Mathematics of Oil Recovery, Amsterdam, The Netherlands, 4-7 September

Caumon, G., Strebelle, S., Caers, J., Journel, G [2004] Assessment of Global Uncertainty for Early Appraisal of Hydrocarbon Fields, SPE 89943, Annual Technical Meeting and Exhibition, Houston, Texas, USA, 26-29 September

Cerv, V., Menvielle, M., Pek, J [2007] Stochastic Interpretation of Magnetotelluric Data, Comparison of Methods, Annals of Geophysics, volume 50, number 1

Chakraborti, N., Kumar, A [2003] The Optimal Scheduling of a Reversing Strip Mill: Studies Using Multi-population Genetic Algorithms and Differential Evolution. Materials and Manufacturing Processes, volume 18, issue 3, 433-445

Chang, C.S., Xu., D.Y, Quek, H.B [1999] Pareto-Optimal Set Based Multiobjective Tuning of Fuzzy Automatic Train Operation for Mass Transit System, IEEE Proceedings on Electric Power Applications, volume 146, number 5, 557-583

Chankong, V., Haimes, Y.Y [1983] Multiobjective Decision Making: Theory and Methodology, North-Holland, New York

Chappell, N.A., Lancaster, J.W [2007] Comparison of Methodological Uncertainties within Permeability Measurement, Hydrological Processes, 21, 2504-2514

Chavent, G., Dupuy, M., Lemonnier, P [1973] History Matching by Use of Optimal Theory, SPE 4627, SPE-AIME 48th Annual Fall Meeting, Las Vegas, USA, Sept 30-October 3

Chebouba, A., Yalaoui, F., Smati, A., Amodeo, L., Younsi, K., Tairi, A [2009] Optimization of Natural Gas Pipeline Transportation Using Ant Colony Optimization, Computers and Operations Research, 36, 1916-1923

Chelouah, R., Siarry, P [1999] Enhanced Continuous Tabu Search. In Voss, S., Martello, S., Osman, I., Roucairol, C. (Eds.), Meta-Heuristics Advances and Trends in

Local Search Paradigms for Optimization, Kluwer Academic Publishers, Boston, MA, 49-61

Chelouah, R., Siarry, P [2000] A Continuous Genetic Algorithm Designed for the Global Optimization of Multimodal Functions, Journal of Heuristics, 6, 191-213

Chen, D., Quirein, J., Hamid, S., Smith, H., Grable, J [2004] Neural Network Ensemble Selection Using Multiobjective Genetic Algorithm in Processing Pulsed Neutron Data, SPWLA 45th Annual Logging Symposium, Noordwijk, The Netherlands, 6-9 June

Chen, S., Heng, L., Yang, D [2009] Production Optimization and Uncertainty Assessment in a CO_2 Flooding Reservoir, SPE 120642, Production and Operations Symposium, Oklahoma City, Oklahoma, USA, 4-8 April

Chen, W., Gavalas, G., Wasserman, M [1973] A New Algorithm for Automatic History Matching, SPE 4545, SPE-AIME 48th Annual Fall Meeting, Las Vegas, USA, 30 September - 3 October

Chen, W., Shi, Y., Teng, H [2008] A Generalized Differential Evolution Combined with EDA for Multi-Objective Optimization Problems, In: Huang et al (Eds.) ICIC, LNAI 5227, 140-147, Springer-Verlag Berlin Heidelberg

Cheong, C.Y., Tan, K.C [2008] A Multi-Objective Multi-Colony Ant Algorithm for Solving the Berth Allocation Problem, Studies in Computational Intelligence, 116, 333-350

Chiampi, M., Fuerntratt, G., Magele, C., Ragusa, C., Repetto, M [1998] Multiobjective Optimization with Stochastic Algorithms and Fuzzy Definition of Objective Function, International Journal of Applied Electromagnetic and Mechanics, IOS Press, volume 9, number 4, 381-389

Choudhary, M., Yoon, S., Ludvigsen, B [2007] Application of Global Optimization Methods for History Matching and Probabilistic Forecasting – Case Studies, SPE 105208, 15th Middle East Oil and Gas Show and Conference, Bahrain, 11-14 March

Chowdhury, S., Dulikravich., Moral, R [2009] Modified Predator-Prey Algorithm for Constrained and Unconstrained Multiobjective Optimization, International Journal of Mathematical Modeling and Numerical Optimization, volume 1, number 1/2, 1-38

Christie, M., Macbeth, C., Subbey, S [2002] Multiple History Matched Models for Teal South, The Leading Edge, March, 286-289

Christie, M., Blunt, M [2001] Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques, SPE 66599, Reservoir Simulation Symposium, Houston, Texas, 11-14 February

Christie, M., Glimm, J., Grove, J.W., Higdon, D.M., Sharp, D.H, Wood-Schultz, M.M [2005] Error Analysis and Simulations of Complex Phenomena, Los Alamos Science, 29

Christie, M., Demyanov, V., Erbas, D [2006] Uncertainty Quantification for Porous Media Flows, Journal of Computational Physics, 217, 143-158

Christie, M., Demyanov, V., Pickup, G.E., O'Sullivan, A., Arnold, D., Erbas, D., Okano, H., Valjak, M [2006-b] Uncertainty Quantification and Geological Parameterization, DEVEX Conference, Aberdeen, UK, 12-13 May

Clark, E., Tunaro, R [2003] Quantification of Political Risk with Multiple Dependent Sources, Journal of Economics and Finance, volume 27, number 1, 125-135

Coats, K., Dempsey, J., Henderson, J [1968] A New Technique for Determining Reservoir Description From Field Performance Data, SPE 2344, 43rd Annual Fall Meeting, Houston, Texas, USA, September 29- October 2

Cobb, B.R., Rumi, R., Salmeron, A [2007] Bayesian Network Models with Discrete and Continuous Variables, Springer Studies in Fuzzy Logic, 213, 81-102

Coello, C.A., Landa Becerra, R [2003] Evolutionary Multiobjective Optimization Using a Cultural Algorithm, IEEE Swarm Intelligence Symposium, Indianapolis, Indiana, USA, 6-13

Coello, C.A., Lamont, G., Van Veldhuizen, D.A [2007] Evolutionary Algorithms for Solving Multiobjective Problems, Second Edition, Springer

Coello, C.A., Toscano, G [2000] A Micro-Genetic Algorithm for Multiobjective Optimization, Technical Report Lania-R1-2000-06, Laboratoria Nacional de Informatica avanzada, Xalapa, Veracruz, Mexico

Cohan, J.L., Marks, D.H [1975] A Review and Evaluation of Multiobjective Programming Techniques, Water Resources Research, volume 11, number 2, 208-220

Confesor, R., Whittaker, G [2007] Automatic Calibration of Hydrologic Models With Multiobjective Evolutionary Algorithm and Pareto Optimization, JAWRA Journal of the American Water Resources Association, volume 43, number 4, 981-989

Coricelli, G., Critchley, H.D., Joffily, M., O'Doherty, J.P., Sirigu, A., Dolan, A.R [2005] Regret and Its Avoidance: A Neuroimaging Study of Choice Behavior, Nature Neuroscience, 8, 1255-1262

Cruz, L. Willigenburgh, V. Van Straten, G [2003] Efficient Differential Evolution Algorithms for Multimodal Optimal Control Problems, Applied Soft Computing, volume 3, number 2, 97-122.

Cua, G [2005] Creating the Virtual Seismologist: Developments in Ground Motion Characterization and Seismic Early Warning, PhD Thesis, California Institute of Technology, USA

Dal Moro, G., Pipan, M [2007] Joint Inversion of Surface Wave Dispersion Curves and Reflection Travel Times via Multiobjective Evolutionary Algorithms, Journal of Applied Geophysics, 61, 56-81

Damsleth, E., Hage, A., Volden, R [1992] Maximum Information at Minimum Cost: Development Study With an Experimental Design, SPE 23139, Journal of Petroleum Technology, December, 1350-1356

Darby, J., Chol-Won, L., Muscatelli, A [2004] Political Uncertainty, Public Expenditure and Growth, European Journal of Political Economy, volume 20, issue 1, 153-179

Das, I., Dennis, J.E [1997] A Closer Look at Drawbacks of Minimizing Weighted Sums of Objectives for Pareto Set Generation in Multicriteria Optimization Problems, Structural Optimization, 14, 63-69

Dashevskiy, D., Dahl, T., Brooks, A., Zurcher, D., Lofts, J., Dankers, S [2006] Dynamic Depth Correction to Reduce Depth Uncertainty and Improve MWD/LWD Log Quality, SPE 103094, Annual Technical Conference and Exhibition, San Antonio, Texas, USA, 24-27 September

Datta, D., Figueira, J. R [2010] A Real-Integer-Discrete-Coded Differential Evolution Algorithm: A Preliminary Study, In Cowling, P and Merz, P (Eds.) 10th European Conference on Evolutionary Computation in Combinatorial Optimization (EvoCOP 2010), LNCS 6022, Springer, 35-46

Davies, B [2010] Limits of Automatic History Matching, Workshop on Conditioning Reservoir Models to Dynamic Data, EAGE/EUROPEC Meeting, Barcelona, Spain, 14-17 June

Dean, R.H., Gai, X., Stone, C.M., Minkoff, S.E [2003] A Comparison of Techniques for Coupling Porous Flow and Geomechanics, SPE 79709, Reservoir Simulation Symposium, Houston, Texas, 3-5 February

Deb, K [2001] Multiobjective Optimization Using Evolutionary Algorithm, Wiley

Deb, K [2008] Introduction to Evolutionary Multiobjective Optimization, In J. Branke et al. (Eds.) Multiobjective Optimization, LNCS 5252, 59-96, Springer-Verlag Berlin Heidelberg

Deb, K., Agrawal, S., Pratap, A., Meyarivan [2000] A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multiobjective Optimization; NSGA-II, Proceedings of the Parallel Problem Solving from Nature VI Conference, 849-858

Demyanov, V [2003] Uncertainty Assessment Using Geostatistics in Reservoir Forecasting: Use of NA on PUNQ-S3 Problem, Progress Report, Uncertainty Quantification Group, Institute of Petroleum Engineering, Edinburgh, UK

Demyanov, V., Subbey, S., Christie, M [2004] Uncertainty Assessment in PUNQ-S3 Neighbourhood Algorithm Framework for Geostatistical Modeling, 9th European Conference on Mathematics of Oil Recovery, Cannes, France, 30 August- 2 September

Deneubourg, J., Aron, S., Goss, S., Pasteels, J [1990] The Self-Organizing Exploratory Pattern of the Argentine Ant, Journal of Insect Behavior, 3, 159-168

Djikpesse, H.A., Couet, B., Wilkinson, D [2010] Gas Lift Optimization Under Facilities Constraints, SPE 136977, Nigeria Annual International Conference and Exhibition, Tinapa-Calabar, Nigeria, 31 July-7 August

Dogru, A.H., Fung, L.S., Middya, U., Al-Shaalan, T., Pita, J., HemmanthKumar, K., Su, H., Tan, J., Hoy, H., Dreiman, W.T., Hahn, W.A., Al-Harbi, R., Al-Youbi, A., Al-Zamed, N., Mezghani, M., Al-Mani, T [2009] A Next Generation Parallel Reservoir Simulator for Giant Reservoirs, SPE 119272, Reservoir Simulation Symposium, Woodland, Texas, USA, 2-4 February

Dorigo M [2004] Ant Colony Optimization and Swarm Intelligence, LNCS 3127, Springer-Verlag, Berlin, Germany

Dorigo M., Stutzle, T [2004] Ant Colony Optimization, MIT Press, ISBN 0-262-04219-3

Dorigo, M [1992] Ottmizzazione, Apprendimento Automatico, ed Algoritmi Basati su Metafora Naturale, PhD Dissertation, Politecnico di Milano, Italy

Dorigo, M [2006] Ant Colony Optimization and Swarm Intelligence, LNCS 4150, Springer-Verlag, Berlin, Germany

Dorigo, M [2008] Ant Colony Optimization and Swarm Intelligence, LNCS 5217, Springer-Verlag, Berlin, Germany

Dorigo, M., Gambardella, L [1997] Ant Colony System: A Cooperative Learning Approach to the Travelling Salesman Problem, IEEE Transactions on Evolutionary Computation, 1, 53-66

Dorigo, M., Maniezzo, V., Colorni, A [1991] Positive Feedback as a Search Strategy, Technical Report No 91-016, Politecnico di Milano, Italy

Dorigo, M., Stutzle, T [2001] An Experimental Study of the Simple Ant Colony Optimization Algorithm, Proceedings of WSES International Conference on Evolutionary Computation (EC'01). WSES Press International, 253-258

Douady, C.J., Delsuc, F., Boucher, Y., Doolittle, W.F., Douzery, E.J.P [2003] Comparison of Bayesian and Maximum Likelihood Bootstrap Measures of Phylogeneic Reliability, Molecular Biology and Evolution, volume 20, issue 2, 248-254

Dougherty, E., Kheirkhah, D [1975] History Matching of Gas Simulation Models Using Optimal Control Theory, SPE 5371, 45th Annual California Regional Meeting of the Society of Petroleum Engineers of AIME, Venture, California, April 2-4

Downton, J.E., Gray., D., Zuk, T [2007] Visualizing AVAZ Parameter Estimates with Uncertainty Due to Noise, EAGE 69th Conference and Exhibition, London, UK, 11-14 June

Dreo, J., Siarry, P [2004] Continuous Interacting Ant Colony Algorithm Based on Dense Hierarchy, Future Generation Computer Systems, volume 20, number 5, 841-856

Dromgoole, P., Speers R [1997] Geoscore – A Method for Quantifying Uncertainty in Field Reserve Estimates, Petroleum Geosciences, volume 3, 1-12

Du, J., McColpin, G.R., Davis, E.J., Marsic, S., Pinnacle, A [2010] Model Uncertainties and Resolution Studies With Application to Subsurface Movement of a CO_2 Injection Project in the Krechba Field Using InSAR Data, SPE 138968, Journal of Canadian Petroleum Technology, volume 49, number 6, 31-37

Dubois, D., Prade, H [1982] On Several Representations of an Uncertain Body of Evidence, In Gupta, M.M., Sanchez, E (Eds.): Fuzzy Information and Decision Processes, 167-181

Dubois, D., Prade, H., Sandri, S [1993] On Possibility/Probability Transformations, In Lowen, R., Roubens, M (Eds.) Fuzzy Logic, 103-112

Eberhart, R., Shi, Y [2001] Particle Swarm Optimization, Developments, Applications and Resources, IEEE Congress in Evolutionary Computation, Seoul, Korea, 27-30 May

Ebrom, D., Krail, P., Ridyard, D., Scott, L [1998] 4-C/4-D at Teal South, The Leading Edge, volume 17, number 10, 1450-453

Edgeworth, F.Y [1881] Mathematical Psychics, P. Keagan, England

Edris, N [2009] Identification of an Appropriate Data Assimilation Approach in Seismic History Matching and It's Effect on Prediction Uncertainty, PhD Thesis, Institute of Petroleum Engineering, Heriot Watt University

Efron, B., Tibshirani, R.J [1993] An Introduction to the Bootstrap, Chapman and Hall

Eichfelder, G [2008] Adaptive Scalarization Methods in Multiobjective Optimization, Vector Optimization series, Springer

Eide, A., Holden, L., Reiso, E., Aanonsen, S [1994] Automatic History Matching by Use of Response Surfaces and Experimental Design, 4th European Conference on the Mathematics of Oil Recovery, Roros, Norway, 7-10 June

Eisenbergh-Klein, G., Pruessmann, J., Gierse, G., Trappe, H [2008] Noise Reduction in 2D and 3D Seismic Imaging by the CRS Method, The Leading Edge, volume 27, number 2, 258-265

Elabed, S [2003] Analysis of the Neighbourhood Algorithm Applied to History Matching Under Uncertainty, Technical Report, Heriot Watt University, Institute of Petroleum Engineering

Elboth, T., Presterud, I., Hermansen, D [2009] Time-Frequency Seismic Data De-Noising, Geophysical Prospecting, volume 58, issue 3, 441-453

Elkins, E [1972] Uncertainty of Oil in Place in Unconsolidated Sands Reservoirs – A Case Study, SPE 3789, Journal of Petroleum Technology, volume 24, number 9, 1315-1319

Emera, M.K., Sarma, H.K [2005] Use of Genetic Algorithm to Predict Minimum Miscibility Pressure (MMP) between Flue Gases and Oil in Design of Flue Gas Injection Project, SPE 93478, Middle East Oil and Gas Show and Conference, Kingdom of Bahrain, 12-15 March

Emerick, A., Silva, E., Messer, B., Luciana, F., Szwarcman, D., Aurelio, M., Pacheco, C [2009] Well Placement Optimization Using a Genetic Algorithm with Nonlinear Constraints, SPE 118808, Reservoir Simulation Symposium, The Woodlands, Texas, USA, 2-4 February

Emerick, A., Reynolds, A.C [2010] EnKF-MCMC, SPE 131375, EUROPEC/EAGE Annual Conference and Exhibition, Barcelona, Spain, 14-17 June

Erbas, D [2006] Sampling Strategies for Uncertainty Quantification in Oil Recovery Prediction, PhD Thesis, Institute of Petroleum Engineering, Heriot Watt University, Edinburgh, UK

ERCH [1998] 4D – 4C Acquisition Consortium Kick – Off Meeting, Teal South Project, Energy Research Clearing House (ERCH), Woodlands, Texas, USA, 17 August

Evensen, G [1994] Sequential Data Assimilation with Nonlinear Quasi-Geostrophic Model Using Monte Carlo Methods to Forecast Error Statistics, Journal of Geophysical Research, 99, 143-162

Fan, Y., Markitell, B.N., Marple, B.D, Valko, P.P [2000] Evaluation of Frac-and-Pack Completions in the Eugene Island, SPE 63107, Annual Technical Conference and Exhibition, Dallas, Texas, USA, 1-4 October

Fasanino, G., Molinard, E., Marsily, G., Pelce, V [1986] Inverse Modeling in Gas Reservoirs, SPE 15592, Annual Technical Conference and Exhibition, New Orleans, October 5-8

Fenwick, D.H., Roggero, F [2003] Updating Stochastic Reservoir Models With New Production Data, SPE 84467, Annual Technical Conference and Exhibition, Denver, Colorado, USA, 5-8 October

Fernandez, J., Echeverria, D., Mukerji, T [2009] Application of Particle Swarm Optimization to Reservoir Modeling and Inversion, IAMG09 Conference, Stanford University, 23-28 August

Floris, F.J.T., Bush, M.D., Cuypers, M., Roggero, F., Syversveen, A [2001] Methods for Quantifying the Uncertainty of Production Forecasts, Petroleum Geoscience, volume 7, issue 10, 87-96

Floris, F.J.T., Peersmann, M.R.H.E [2002] Integrated Scenario and Probabilistic Analysis for Asset Decision Support, Petroleum Geoscience, volume 8, 1-6

Fonseca, C.M., Fleming, P.J [1993] Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In: Proceedings of the Fifth International Conference on Genetic Algorithms, 416-423

Franssen, H., Kinzelbach, W [2009] Ensemble Kalman Filtering Versus Sequential Self-Calibration for Inverse Modeling of Dynamic Groundwater Flow Systems, Journal of Hydrology, volume 365, 261-274

Friedel, T., Tewari, R., Rasidi, T.G., Flew, S., Strasser, R., Trebolle, R., Belfield, W., Caretta, F [2009] Efficient Uncertainty Management Workflow for Brownfield Redevelopment, SPE 125673, SPE/EAGE Reservoir Characterization and Simulation Conference, Abu Dhabi, UAE, 19-21 October

Friedemann, J., Varma, A., Bonissone, P., Iyer, N [2008] Subsea Condition Monitoring: A Path to Increased Availability and Increased Recovery, SPE 112051, Intelligent Energy Conference and Exhibition, Amsterdam, The Netherlands, 25-27 February

Frizzell, D., Sibley, M., Cotner, B., McCartney, S., Schmidt, G., Burkers, R., Phelps, J., Tosdevin, M., Mazloom, J [2007] Modeling Uncertainties of a Gas-Delivery Project, IPTC 11540, International Petroleum Technology Conference, Dubai, U.A.E, 4-6 December

Fu, J., Fenton, R., Cleghorn, W [1991] A Mixed Integer-Discrete-Continuous Programming Method and Its Application to Engineering Design Optimization, Engineering Optimization, volume 17, number 4, 263-280

Fuchs, M., Neumaier, A [2008] Handling Uncertainty in Higher Dimensions with Potential Clouds towards Robust Design Optimization, In Dubois, D et al. (Eds.): Soft Methods for Handling Variability and Imprecision, Springer-Verlag Berlin Heidelberg, 376-382

Galli, A., Armstrong, M., Dias, M [2004] The Value of Information: A Bayesian Real Option Approach, SPE 90418, Annual Technical Meeting and Exhibition, Houston, Texas, USA, 26-29 September

Gambardella, L.M., Taillard, E., Agazzi, G [1999] MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows, In Corne, D., Dorigo, M., Glover, F (Eds.) New Ideas in Optimization, 63-76, London, McGraw-Hill

Gao, G., Zafari, M., Reynolds, C.A [2005] Quantifying Uncertainty for the PUNQ-S3 Problem in a Bayesian Setting with RML and EnKF, SPE 93324, Reservoir Simulation Symposium, Houston, Texas, USA, 31 January – 2 February

Gao, G., Li, G., Reynolds, A.C [2007] A Stochastic Optimization Algorithm for Automatic History Matching, SPE Journal, volume 12, number 2, 196-208

Gao, W [2007] Study on Immunized Ant Colony Optimization, Third International Conference on Natural Computation (ICNC), 792-796

Gao, J., Wang, J [2009] WBMOAIS: A Novel Artificial Immune System for Multiobjective Optimization, Journal of Computers and Operations Research, 37, 50-61

Garb, A [1988] Assessing Risk in Estimating Hydrocarbon Reserves and Evaluating Hydrocarbon Producing Properties, SPE 15921, Journal of Petroleum Technology, volume 40, number 6, 756-778

Gavalas, G.R, Shah, P.C., Seinfeld, J.H [1976] Reservoir History Matching by Bayesian Estimation, SPE 5740, SPE Journal, volume 16, number 6, 337-350

Gazet, S., Chavanne, E., Khalaf, A [2009] Evaluating the Impact of Geophysical Uncertainties on Gross Rock Volume in a Low Relief Carbonate Field, Offshore Abu Dhabi, SPE 125573, SPE/EAGE Reservoir Characterization and Simulation Conference, Abu Dhabi, UAE, 19-21 October

Geoffrion, A.M., Dyer, J.S., Feinberg, A [1972] An Interactive Approach for Multi-Criterion Optimization with an Application the Operation of an Academic Department, Management Science, 19, 257-369

Glen Van Horn, H [1970] Gas Reserves, Uncertainty and Investment Decisions, SPE 2878, Gas Industry Symposium, Omaha, Nebraska, USA, 21-22 May

Glimm, J., Hou, S., Lee, Y., Sharp, D., Ye, K [2001] Prediction of Oil Production with Confidence Intervals, SPE 66350, Reservoir Simulation Symposium, Houston, Texas, USA, 11-14 February

Glover, F [1977] Heuristics for Integer Programming Using Surrogate Constraints, Decision Sciences, 8, 156-166

Goldberg, D [1985] Computer-Aided Gas Pipeline Operation Using Genetic Algorithm and Rule Learning: Part 1 Genetic Algorithm in Pipeline Optimization, SPE 14590, SPE Journal, Unsolicited Manuscript

Goldberg, D.E [1989] Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Publishing Company, Reading

Gong, W., Zhihua, C., Zhu, L [2009] An Efficient Multiobjective Differential Evolution Algorithm for Engineering Design, Struct Multidisc Optim, 38, 137-157

Gong, Y., Xu, R., Zhang, J [2009 b] A Clustering-Based Adaptive Parameter Control Method for Continuous Ant Colony Optimization, Proceedings of the 2009 IEEE International Conference on Systems, Man and Cybernetics, San Antonio, Texas, USA, 1827-1832

Goss S, Aron S, Deneubourg L, Pasteels M [1989] Self-Organized Shortcuts in the Argentine Ant, Naturwissenschaften, 76, 579-581

Gosselin, O., Anonsen, S.I., Aavatsmark, I., Cominelli, A., Gonard, R., Kolasinski, M., Ferinandi, F., Kovacic, L., Neylon, K [2003] History Matching Using Time-Lapse Seismic (HUTS), SPE 84464, Annual Technical Conference and Exhibition, Denver, Colorado, 5-8 October

Gravel, M., Price, W.L., Gagne, C [2002] Scheduling Continuous Casting of Aluminum Using a Multiple-Objective Ant Colony Optimization Metaheuristic, European Journal of Operations Research, volume 143, number 1, 218-229

Gu, Y., Oliver, D [2005] History Matching of the PUNQ-S3 Reservoir Model Using the Ensemble Kalman Filter, SPE 89942, SPE Journal, volume 10, number 2, 217-224

Guntsch, M [2004] Ant Algorithms in Stochastic and Multi-Criteria Environments, PhD Thesis, Department of Economics and Business Engineering, University of Karlsruhe, Germany

Haase, O., Schulze-Riegert, R., Junker, Plas, L [2006] Selecting Optimal Cases for Uncertainty Quantification and History Matching, Oil and Gas European Magazine, 4

Hackel, S., Fischer, M., Zechel, D [2008] A Multi-Objective Ant Colony Approach for Pareto-Optimization Using Dynamic Programming, GECCO, Atlanta, Georgia, USA, 33-40

Hadidi, R [2007] Generic Probabilistic Inversion Technique for Geotechnical and Transportation Engineering Applications, PhD Thesis, Rutgers University, NJ

Hadidi, R., Gucunski, N [2009] Probabilistic Inversion: A New Approach to Inversion Problems in Pavement and Geomechanical Engineering, In Intelligent and Soft Computing in Infrastructure Systems Engineering, K Gopalakrishnan et al. (Eds.) 21-45, Springer

Haimes, Y.Y., Lasdon, L.S, Wismer, D.A [1971] On a Bicriterion Formulation of the Problems of Integrated System Identification and System Optimization, IEEE Transactions on Systems, Man, and Cybernetics, volume 1, number 3, 296-297

Hajela, P., Lin, C [1992] Genetic Search Strategies in Multicriterion Optima; Design, Structural Optimization, 4, 99-107

Hajizadeh, Y. Christie, M. and Demyanov, V [2009] Ant Colony Optimization for History Matching, SPE 121193, EUROPEC/EAGE Annual Meeting, Amsterdam, The Netherlands, 8-11 June.

Hamill, T., Whitaker, J., Snyder, C [2009] Distance-Dependent Filtering of Back-Ground Error Covariance Estimates in an Ensemble Kalman Filter, Mon Wea Rev, 129, 2760-2790

Han, Y.M., Park, C., Kang, K.M [2010] Estimation of Future Production Performance Based on Multi-objective History Matching in a Waterflooding Project, SPE 130500, EUROPEC/EAGE Annual Conference and Exhibition, Barcelona, Spain, 14-17 June

Hansen, N., Ostermeier, A [2001] Completely Derandomized Self-Adaption in Evolution Strategies, Evolutionary Computation, volume 2, number 9, 159-195

Harrison, G.H., Tweedie. J.A [1981] A Multiobjective Economic Analysis of Oilfield Production Policy, SPE 10327, 56th Annual Fall Technical Conference and Exhibition, Texas, USA, 5-7 October

Hart, E., McEwan, C., Timms, J., Andy Hone (Eds.) [2010] Proceedings of 9th International Conference on Artificial Immune Systems, ICARIS 2010, Springer LNCS 6209, Edinburgh, UK

Henderson, N., Sacco, W.F., Barufatti, N.E., Ali, M.M [2010] Calculation of Critical Points of Thermodynamic Mixtures with Differential Evolution Algorithms, Ind. Eng. Chem. Res., volume 49, issue 4, 1872-1882

Hernandez-Diaz, A.G., Coello, C., Perez, F., Caballero, R., Molina, J., Santana-Quintero, L.V [2008] Seeding the Initial Population of a Multi-Objective Evolutionary Algorithm Using Gradient-Based Information, In IEEE Congress on Evolutionary Computation (CEC 2008), Hong Kong, 1617-1624

Heward, A.P., Gluyas, J.C [2002] How Can We Help Ensure Success of Oil and Gas Field Rehabilitation Projects? Petroleum Geoscience, volume 8, 299-306

Holland, J [1975] Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor

Holldobler, B., Wilson, E [1990] “The Ants”, Cambridge, MA, Harvard University Press

Horn, J., Nafpliotis, N., Goldberg, D [1994] A Niched Pareto Genetic Algorithm for Multiobjective Optimization, In Proceedings of the First IEEE Conference on Evolutionary Computation, volume 1, Piscataway, NJ, 119-124

Hornburger, G.M., Spear, R.C [1981] An Approach to the Preliminary Analysis of Environmental Systems, Journal of Environmental Management, 12, 7-18

Horne, R [1994] Uncertainty in Well Test Interpretation, SPE 27972, University of Tulsa Centennial Petroleum Engineering Symposium, Tulsa, Oklahoma, USA, 29-31 August

Hu, X., Chang, J., Li, Y [2008] Orthogonal Methods Based Ant Colony Search for Solving Continuous Optimization Problems, Journal of Computer Science and Technology, 23, 2-18

Huang, V.L., Suganthan, P.N., Qin, A.K, Baskar, S [2005] Multiobjective Differential Evolution with External Archive and Harmonic Distance-Based Diversity Measure, Technical Report, School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore

Huettel, S.A., Song, A.W., McCarthy, G [2005] Decisions Under Uncertainty: Probabilistic Context Influence Activation of Prefrontal and Parietal Cortices, The Journal of Neuroscience, volume 25, issue 13, 3301-3311

International Energy Agency (IEA) [2008] World Energy Outlook, Paris, France

Iorio, A.W., Li, X [2004] Solving Rotated Multiobjective Optimization Problem Using Differential Evolution, In: Webb, G.I., Yu, X. (Eds.) AI 2004. LNCS Springer, volume 3339, 861-872

Iqbal, A., Dar, N [2009] A Self Progressing Fuzzy Rule-Based System for Optimizing and Predicting Machining Process, In Ao and Gelman (Eds.) Advances in Electrical Engineering and Computational Science, Springer Lecture Notes in Electrical Engineering, Volume 39, 439-446

Iredi, S., Merkle, D., Middendorf, M [2001] Bi-Criterion Optimization with Multi Colony Ant Algorithm, In Zitzler, E (et al) (Eds.) Proceedings of the First International Conference on Evolutionary Multi-Criterion-Optimization (EMO'01), LNCS 1993, 359-372

Irgens, M., Lasher, W [2007] Use of Advanced Optimization Techniques to Manage a Complex Drilling Schedule, SPE 110805, Annual Technical Conference and Exhibition, Anaheim, California, USA, 11-14 November

ISO [2003] International Vocabulary of Basic and General Terms in Metrology, International Organization for Standardization (ISO), Geneva, Switzerland, ISBN 92-67-10175-1

Iwegbue, M., Appah, D., Ogwo, E.A [2007] Crude Oil Metering Experience in the Niger-Delta, SPE 111906, Nigeria Annual International Conference and Exhibition, Abuja, Nigeria, 6-8 August

Izgec, B., Cribbs, M.E., Pace, S.V, Zhu, D., Kabir, C.S [2007] Placement of Permanent Downhole Pressure Sensors in Reservoir Surveillance, SPE 107268, EUROPEC/EAGE Conference and Exhibition, London, UK, 11-14 June

Jacquard, P., Jain, C [1965] Permeability Distribution from Field Pressure Data, SPE 1307, SPE Journal, volume 5, number 4, 281-294

Jahangiri, H.R [2007] Production Optimization Using Smart Well Technology with Differential Evolution Algorithm, Graduate Student Symposium, University of Southern California

James, B., Kerr, A., Lim, S., Lewandowski, E., Knight, C., Bell, R [2008] Intelligent Fields Management at Woodside: A Low-Cost Step Improvement in Field Management Using Off the Shelf Technology, SPE 116519, Asia Pacific Oil and Gas Conference and Exhibition, Perth, Australia, 20-22 October

Jansky, J., Plicka, V., Novotny, O [2007] Determining a 1-D Velocity Model of the Uppermost Crust from P and S Arrival Times Using the Neighbourhood Algorithm, Synthetic Test, Acta Geodyn, Geomater, volume 4, number 2, 146, 5-12

Jardak, M., Navan, M., Zupanski, M [2009] Comparison of Sequential Data Assimilation Methods for the Kuramoto-Sivashinsky Equation, International Journal for Numerical Methods in Fluids, volume 62, issue 4, 374-402

Jayaraman, V., Kulkarni, B., Karale, S. and Shelokar, P [2000] Ant Colony Framework for Optimal Design and Scheduling of Batch Plants, Computer and Chemical Engineering, 24, 1901-1912

Ji, Z., Xie, Z [2008] Multi-Objective Optimization of Continuous Casting Billet Based on Ant Colony System Algorithm, IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application, 262-266

Jia, X., Cunha, L.B., Deutsch, C.V [2009] Investigation of Stochastic Optimization Methods for Automatic History Matching of SAGD Processes, Journal of Canadian Petroleum Technology, volume 48, 14-18

Jiang H, Deng H, He Y [2008] Determination of Fuzzy Logic Membership Functions Using Extended Ant Colony Optimization Algorithm, Fifth International Conference on Fuzzy Systems and Knowledge Discovery, Jinan, Shandong, China, 18-20 October

Jin, Y [2005] A Comprehensive Survey of Fitness Approximation in Evolutionary Computation, Soft Computing, 9, 1, 3-12

Joshi, R., Sanderson, A [1999] Minimal Representation Multisensor Fusion Using Differential Evolution, IEEE Transactions on Systems, MAN, and Cybernetics, volume 29, number 1, 63-76

Journel, A., Bitanov, A [2004] Uncertainty in N/G Ratio in Early Reservoir Development, Journal of Petroleum Science and Engineering, 44, 2004, 115-130

Jutila, H., Goodwin, N [2006] Schedule Optimization to Complement Assisted History Matching and Prediction Under Uncertainty, SPE 100253, EUROPEC Conference and Exhibition, Vienna, Austria, 12-15 June

Karaboga, N. Cetinkaya, B [2004] Performance Comparison of Genetic Algorithm and Differential Evolution Algorithms for Digital FIR Filter Design, In Advances in Information Systems, Springer, 482-488

Karaboga, D [2005] An Idea Based on Honey Bee Swarm for Numerical Optimization. Technical Report, TR06, Department of Computer Science, Engineering Faculty, Erciyes University, Turkey

Karimi, A., Nobahari, H., Siarry, P [2010] Continuous Ant Colony System and Tabu Search Algorithms Hybridized for Global Minimization of Continuous Multi-Minima Functions, Computer Optimization Applications, 45, 639-661

Kathrada, M [2009] Uncertainty Evaluation of Reservoir Simulation Models Using Particle Swarm and Hierarchical Clustering, PhD Thesis, Institute of Petroleum Engineering, Heriot Watt University, Edinburgh, UK

Kennedy, J., Eberhart, R [1995] Particle Swarm Optimization, Proceedings of IEEE International Conference on Neural Networks, volume 4, 1942-1948

Kennett, B [2006] Non-Linear Methods for Event Location in a Global Context, Physics of the Earth and Planetary Interiors, 158, 46-54

Kern, S., Muller, S., Hansen, N., Buche, D., Ocenasek, J., Koumoutsakos, P [2004] Learning Probability Distributions in Continuous Evolutionary Algorithms – A Comparative Review, Natural Computing, volume 3, number 1, 77-112

Khalidji, M., Zeiaee, M., Taei, A., Jahed-Motlagh, M.R [2009] Dynamically Weighted Continuous Ant Colony Optimization for Bi-Objective Portfolio Selection Using Value-at-Risk, 3rd Asia International Conference on Modeling and Simulation, 230-235

Khemka, N., Jacob, C [2010] VISPLORE: Exploring Particle Swarms by Visual Inspection, In Sarker and Ray (Eds.) Agent-Based Evolutionary Search, Springer, ALO 5, 255-284

Kim, I.Y., de Weck, O.L [2006] Adaptive Weighted Sum Method for Multiobjective Optimization: A New Method for Pareto Front, Structural and Multidisciplinary Optimization, volume 31, number 2, 105-116

Klir, J.G [2006] Uncertainty and Information: Foundations of Generalized Information Theory, Wiley- Interscience

Knowles, J., Corne, D.W [2000] Approximating the Non-dominated Front Using the Pareto Archived Evolution Strategy, Evolutionary Computation Journal, volume 8, number 2, 149-172

Koh, A [2007] Solving Transportation Bi-Level Programs with Differential Evolution, Proceedings of IEEE Congress on Evolutionary Computation, Singapore, September 25-28, 2243-2250

Kong, M., Tian, P [2006] Direct Application of Ant Colony Optimization to Function Optimization Problems in Continuous Domain, In Proceedings of ANTS 2006, Brussels, Belgium, LNCS 4150, 324-331

Koppen, M [2004] No-Free-Lunch Theorems and the Diversity of Algorithms, IEEE Congress on Evolutionary Computation, CEC, 19-23 June 2004, Portland, USA

Kovarik, O [2006] Ant Colony Optimization for Continuous Problems, Master Thesis, Faculty of Electrical Engineering, Czech Technical University in Prague

Kruger, W.D [1960] Determining Areal Permeability Distribution by Calculations, SPE 1580, 35th Annual Fall Meeting, Denver, USA, 2-5 October

Kuczera, G., Parent, E [1998] Monte Carlo Assessment of Parameter Uncertainty in Conceptual Catchment Models, The Metropolis Algorithm, Journal of Hydrology, 211, 69-85

Kukkonen, S., Lampinen, J [2004] An Extension of Generalized Differential Evolution for Multiobjective Optimization with Constraints, In: Yao, X,, Burke, E.K., Lozano, J.A., Smith, J., Merelo-Giervos, J.J., Bullinaria, J.A., Rowe, J.E., Tino, P., Kaban, A., Schewefel, (Eds.) PPSN 2004, LNCS, volume 3242, 752-761

Kukkonen, S., Lampinen, J [2007] Ranking-Dominance and Many-Objective Optimization, IEEE Congress on Evolutionary Computation, CEC 2007, 3983-3990

Kumar, A., Oballa, V., Card, C.C [2010] Fully-Coupled Wellbore Design and Optimization for Thermal Operations, SPE 137427, Canadian Unconventional Resources and International Petroleum Conference, Calgary, Canada, 19-21 October

Kuo R, Chen K [2005] Developing a Diagnostic System Through Integration of Fuzzy Case-Based Reasoning and Fuzzy Ant Colony System, Expert Systems with Applications, 28, 738-797

Kylling, O [2009] Optimizing Separator Pressure in a Multistage Crude Oil Production Plant, Master Thesis, Department of Engineering Cybernetics, Norwegian University of Science and Technology (NTNU)

Lagaros, N., Papadrakakis, M., Plevris, V [2005] Multiobjective Optimization of Space Structures Under Static and Seismic Loading Conditions, In Evolutionary Multiobjective Optimization, Theoretical Advances and Applications, Springer

Lampinen, J., Zelinka, I [1999] Mixed-Integer-Discrete-Continuous Optimization by Differential Evolution, Part 1, The Optimization Method, In Osmere, P. (Ed.) Fifth International Mendel Conference on Soft Computing (DEMDEL, 1999), Brno, Czech Republic, 71-76

Lampinen, J [2001] DE's Selection Rule for Multiobjective Optimization, Technical Report, Lappeenranta University of Technology, Department of Information Technology

Lasrado, V [2009] Digital Oilfield Projects: A Perspective Using Examples from Reservoir and Reserves Management, SPE 124404, Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 4-7 October

Lawniczak, W., Hanea, R., Heemink, A., McLaughlin, D., Jansen, J [2008] History Matching Using a Multiscale Ensemble Kalman Filter, 11th European Conference on the Mathematics of Oil Recovery, Bergen, Norway, 8-11 September

Lawrence, J.J., Gupta, D.K [2009] Quality Assessment and Consistency Evaluation of Hydrocarbon PVT data, IPTC 13784, International Petroleum Technology Conference, Doha, Qatar, 7-9 December

Lazinica, A [2009] Particle Swarm Optimization, InTech Educational Publishing, ISBN 978-953-7619-48-0

Lee, Z., Su, S., Chuang, C., Liu, K [2008] Genetic Algorithm with Ant Colony Optimization (GA-ACO) for Multiple Sequence Alignment, Applied Soft Computing, 8, 58-78

Lee, N.M.Y., Lau, H.Y.K., Ko, A.W.Y [2009] An Immune Inspired Algorithm for Solving Dynamic Vehicle Dispatching Problem in a Port Container Terminal, In

Andrews et al (Eds.) Proceedings of 8th International Conference on Artificial Immune Systems, ICARIS 2009, Springer LNCS 5666, York, UK

Leijen, V., Hermand, J.P [2006] Geoacoustic Inversion and Uncertainty Analysis with Max-Min Ant System, In Dorigo et al. (Eds.), ANTS 2006, LNCS 4150, 420-427

Li, H., Chou, C [1994] A Global Approach for Nonlinear Mixed Discrete Programming in Design Optimization, Engineering Optimization, 22, 109-122

Li, H., Zhang, Q [2006] A Multiobjective Differential Evolution Based on Decomposition for Multiobjective Optimization with Variable Linkages, In Runarsson et al. (Eds) PPSN IX, LNCS 4193, 583-592

Li, Y., Hilton, A [2005] Reducing Spatial Sampling in Long-Term Groundwater Monitoring Networks Using Ant Colony Optimization, International Journal of Computational Intelligence Research, 1, 19-28

Li, Y., Liu, J., Wang, Y [2009] An Adaptive Weighted PSO for Rolling Schedules Multiobjective Optimization for Tandem Cold Rolling, Proceedings of the IEEE International Conference on Automation and Logistics, Shenyang, China

Lin, J [2001] On Min-Norm and Min-Max Methods of Multiobjective Optimization, Mathematical Programming, volume 103, number 1, Springer

Lin, Y., Cai, H., Xiao, J., Zhang, J [2007] Pseudo Parallel Ant Colony Optimization for Continuous Functions, International Conference on Natural Computation (ICNC)

Lin, Y., Zhang, J., Lan, L [2008] A Contour Method in Population-Based Stochastic Algorithms, IEEE World Congress on Computational Intelligence, 1-6 June, Hong Kong

Litvak, M., Christie, M., Johnson, D., Colbert, J., Sambridge, M [2005] Uncertainty Estimation in Production Predictions Constrained by Production History and Time-

Lapse Seismic in a GOM Oil Field, SPE 93146, Reservoir Simulation Symposium, Houston, USA, 31 January – 2 February

Litvak, M., Patrick, A [2009] Field Development Optimization Applied to Giant Oil Fields, SPE 118840, Reservoir Simulation Symposium, The Woodlands, Texas, USA, 2-4 February

Liu, J., Lampinen, J [2005] A Fuzzy Adaptive Differential Evolution Algorithm, Soft Comput Fusion Found Methodol Appl, volume 9, issue 6, 448-462

Liu, N [2005] Automatic History Matching of Geological Facies, PhD Thesis, University of Oklahoma, USA

Liu, N., Oliver, D [2005] Critical Evaluation of the Ensemble Kalman Filter on History Matching of Geological Facies, SPE 92816, Reservoir Simulation Symposium, Houston, USA, 31 January- 2 February

Loh, H. and Papalambros, P [1991] Computation Implementation and Test of a Sequential Linearization Approach for Solving Mixed-Discrete Nonlinear Design Optimization, Journal of Mechanical Design, volume 113, issue 3, 335-345

Lorentzen, R., Naevdal, G., Valles, B., Berg, M., Grimstad, A [2005] Analysis of the Ensemble Kalman Filter for Estimation of Permeability and Porosity in Reservoir Models, SPE 96375, Annual Technical Conference and Exhibition, Dallas, Texas, USA, 9-12 October

Ma, X., Datta-Gupta, A., Efendiev, Y [2008] A Multistage MCMC Method with Nonparametric Error Model for Efficient Uncertainty Quantification in History Matching, SPE 115911, Annual Technical Conference and Exhibition, Denver, Colorado, USA, 21-24 September

Macary, S., Ball, B., Baramawy, M [2005] Can Large Domain of Uncertainty be Reduced for Two One-Well Fault Blocks? SPE 95989, Annual Technical Conference and Exhibition, Dallas, Texas, USA, 9-12 October

Macbeth, C., Shams, A [2006] Observation of Azimuthal Anisotropy from the Seismic Reflectivity of a Tertiary Turbidite Sand, Geophysical Prospecting, 54, 553-564

Mackie, S.I., Begg, S.H., Smith, C., Welsh, M.B [2008] Real World Decision-Making in the Upstream Oil and Gas Industry; Prescriptions for Improvement, APPEA Journal, volume 48, part 1, 329-343

Madadgar S, Afshar A [2009] An Improved Continuous Ant Algorithm for Optimization of Water Resource Problems, Water Resource Management, 23, 2119-2139

Madavan, N.K [2002] Multiobjective Optimization Using a Pareto Differential Evolution Approach, In: Congress on Evolutionary Computation (CEC 2002), Piscataway, New Jersey, volume 2, 1145-1150

Madavan, N.K [2004] On Improving Efficiency of Differential Evolution for Aerodynamic Shape Optimization Applications, 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, September, Albany, NY, USA

Manceau, E., Mezghani, M., Zabala, I., Roggero, F. [2001] Combination of Experimental Design and Joint Modeling Methods for Quantifying the Risk Associated with Deterministic and Stochastic Uncertainties - An Integrated Test Study, SPE 71620, Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 30 September-3 October

Mannon, R [1964] Some Aspects of Production Forecasting, Rocky Mountain Petroleum Section Regional Meeting, Casper, USA, 25-26 May 1964

Mantica, S., Cominelli, A., Mantica, G. [2002] Combining Global and Local Optimization Techniques for Automatic History Matching Production and Seismic Data, SPE 78353, SPE Journal, Volume 7, Number 2, 123-130

Mariano, C.E., Morales, E [1999] MOAQ: An Ant-Q Algorithm for Multiple Objective Optimization Problems, In Banzhaf et al. (Eds.) Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), volume 1, 894-901

Marsily, G., Lavedan, G., Boucher, M., Fasanino, G [1987] Interpretation of Inference Tests in a Well Field Using Geostatistical Techniques to Fit the Permeability Distribution in a Reservoir Model, Geostatistics for Natural Resources Characterization, Part 2, Reidel Publishing Company, 831-849

Marson-Pidgeon, K [2001] Seismogram Synthesis for Teleseismic Events with Application to Source and Structural Studies, PhD Thesis, Australian National University

Martinius, A.W., Naess, A [2005] Uncertainty Analysis of Fluvial Outcrop Data for Stochastic Reservoir, Petroleum Geoscience, volume 11, 203-214

Mauris, G [2008] Inferring a Possibility Distribution from Very Few Measurements, In Dubois, D et al. (Eds.): Soft Methods for Handling Variability and Imprecision, Springer-Verlag Berlin Heidelberg, 92-99

McMichael, C [1999] The Fallacy of the Hockey Stick Projection, SPE 56454, Annual Technical Conference and Exhibition, Houston, Texas, USA, 3-6 October

McMullen, P.R [2001] An Ant Colony Optimization Approach to Addressing a JIT Sequencing Problem with Multiple Objectives, Artificial Intelligence in Engineering, volume 15, number 3, 309-317

McPhee, C., Arthur, K [1994] Relative Permeability Measurements – An Inter-Laboratory Comparison, SPE 28826, European Petroleum Conference, London, UK, October 25-27

Medaglia, A [2003] An Evolutionary Algorithm for Project Selection Problems Based on Stochastic Multiobjective Linearly Constrained Optimization, In Models and

Methods for Project Selection: Concepts from Management Sciences, Finance, and Information Technology, Boston, Kluwer Academic Publishers

Mekapati, S [2005] A Study of Differential Evolution and Tabu Search for Benchmark and Phase Stability Problems, AICHE Annual Meeting, Cincinnati, OH, USA, 30 October- 4 November

Mendiburu, A., Miguel-Alonso, J., Lozano, J.A [2010] A Review of Parallel Estimation of Distribution Algorithms, In Fernandez de Vega, F., Cantu-Paz, E (Eds.) Parallel and Distributed Computational Intelligence, SCI 269, Springer, 143-163

Meyer, M.A., Booker, J.M [2001] Eliciting and Analyzing Expert Judgment, ASA-SIAM Seried on Statistics and Applied Probability, ISBN: 0-89871-474-5

Mezura-Montes, E., Reyes-Sierra, M., Coello, C [2008] Multi-Objective Optimization Using Differential Evolution: A Survey of the State-of-the-Art, U.K. Chakraborty (Ed.): Advances in Differential Evolution, SCI 143, Springer, 173-196

Miettinen, K [1999] Nonlinear Multiobjective Optimization, Kluwer Academic Publishers, Boston

Miettinen, K., Ruiz, F., Wierzbicki, A.P [2008] Introduction to Multiobjective Optimization Interactive Approaches, In Branke et al. (Eds): Multiobjective Optimization, LNCS 5252, 27-57, Springer-Verlag Berlin Heidelberg

Milliken, W., Marjorie, L., Strebelle, S., Zhang, Y [2008] The Effect of Geologic Parameters and Uncertainties on Subsurface Flow: Deepwater Depositional Systems, SPE 114099, Western Regional and Pacific Section AAPG Joint Meeting, Bakersfield, California, USA, 31 March – 2 April

Mohaghegh, S [2000] Virtual Intelligence and Its Applications in Petroleum Engineering - Part 3: Fuzzy Logic, SPE 62415, Distinguished Author Series, Journal of Petroleum Technology, November

Mohaghegh, S [2006] Quantifying Uncertainties Associated with Reservoir Simulation Studies Using Surrogate Reservoir Models, SPE 102492, Annual Technical Conference and Exhibition, San Antonio, Texas, USA, 24-27 September

Mohaghegh, S., Modavi, V., Hafez, H., Haajizadeh, M., Kenawy, M., Guruswamy., S [2006] Development of Surrogate Reservoir Models (SRM) for Fast Track Analysis of Complex Reservoirs, SPE 99667, Intelligent Energy Conference and Exhibition, Amsterdam, Netherlands, 11-13 April

Mohamed, L., Christie, M., Demyanov, V [2009] Comparison of Stochastic Sampling Algorithms for Uncertainty Quantification, SPE 119139, Reservoir Simulation Symposium, The Woodlands, Texas, USA, 2-4 February

Moller, B., Graf, W., Beer, M., Sickert, J [2002] Fuzzy Randomness – Towards a New Modeling of Uncertainty, Fifth World Congress on Computational Mechanics, Vienna, Austria, July 7-12

Monmarche N, Venturini G, Slimane M [2000] On How Pachycondyla Apicalis Ants Suggest a New Search Algorithm, Future Generation Computer Systems, 16, 937-946

Montano, A., Coello, C., Mezura-Montes, E [2010] pMODE-LD+SS: An Effective and Efficient Parallel Differential Evolution Algorithm for Multiobjective Optimization, In Proceedings of the 11th International Conference on Parallel Problem Solving from Nature, Krakow, Poland, 11-15 September

Moore, J.S., Clark, G.C [1988] History Match of the Maljamar CO_2 Pilot Performance, SPE 17323, Enhanced Oil Recovery Symposium, Tulsa, USA, 17-20 April

Moore, R.E [1966] Interval Analysis, Prentice-Hall, Eaglewood Cliffs, N.J., USA

Morales, A., Nasrabadi, H., Zhu, D [2010] A Modified Genetic Algorithm for Horizontal Well Placement Optimization in Gas Condensate Reservoirs, SPE 135182, Annual Technical Conference and Exhibition, Florence, Italy, 19-22 September

Mostaghim, S., Teich, J [2003] Strategies for Finding Good Local Guides in Multiobjective Particle Swarm Optimization (MOPSO), IEEE Swarm Intelligence Symposium Proceedings, Indianapolis, Indiana, USA, 26-33

Murugappan, A [2005] Total Completion Time Minimization in a Drilling Sequence Problem Considering Tool Wear, An Ant Algorithm Approach, Master of Science Thesis, College of Engineering, Wichita State University

Naevdal G [2003] Reservoir Monitoring and Continuous Model Updating Using Ensemble Kalman Filter, SPE 84372, Annual Technical Conference and Exhibition, Denver, Colorado, USA, 5-8 October

Naevdal, G., Mannseth, T., Vefring, E [2002] Near Well Reservoir Monitoring Through Ensemble Kalman Filter, SPE 75235, SPE/DOE Improved Oil Recovery Symposium, Tulsa, Oklahoma, USA, 13-17 April

Nazareth, L [1977] A Conjugate Direction Algorithm without Line Searches, Journal of Optimization Theory Applications, 23, 373-87

Nghiem, L., Chadong, Y., Shrivatava, V,, Kohse, B., Hassem, M,, Chen, D., Card, C [2009] Optimization of Residual Gas and Solubility Trapping for CO_2 Storage in Saline Aquifers, SPE 119080, Reservoir Simulation Symposium, The Woodlands, Texas, USA, 2-4 February

Nguyen, X., Dahm, T., Grevemeyer, I [2009] Inversion of Scholte Wave Dispersion and Waveform Modeling for Shallow Structure of the Ninetyeast Ridge, Journal of Seismology, 13, 543-559

Nicotra, G., Godi, A., Cominelli, A., Christie, M [2005] Production Data and Uncertainty Quantification: A Real Case Study, SPE 93280, Reservoir Simulation Symposium, Houston, USA, 31 January-2 February

Nikravesh, M., Aminzadeh, F., Zadeh, L., Editors [2003] Soft Computing and Intelligent Data Analysis in Oil Exploration, in Series of Developments in Petroleum Science, Elsevier

Nissen, V., Propach, J [1998] On the Robustness of Population-Based Versus Point-Based Optimization in the Presence of Noise, IEEE Transactions on Evolutionary Computation, volume 2, number 3, 107-119

Nogueira, P., Schiozer, D [2009] An Efficient Methodology of Production Strategy Optimization Based on Genetic Algorithms, SPE 122031, Latin American and Caribbean Petroleum Engineering Conference, Cartagena, Colombia, 31 May- 3 June

Ochoa, A [2010] Opportunities for Expensive Optimization with Estimation of Distribution Algorithms

Ocenasek, J., Schwarz, J [2002] Estimation Distribution Algorithm for Mixed Continuous-Discrete Optimization Problems, In Proceedings of 2nd Euro-International Symposium on Computational Intelligence, IOS Press, Amsterdam, Netherlands, 227-232

O'Dell, M., Lamers, E [2005] Subsurface Uncertainty Management and Development Optimization in the Harweel Cluster, South Oman, SPE 89110, SPE Reservoir Evaluation and Engineering, volume 8, number 2, 164-168

Offshore Magazine [2010] Gulf of Mexico Map, January 2010, available at http://www.offshore-mag.com/index/maps-posters.html, last accessed August 2010

Okano, H., Pickup, G., Christie, M., Subbey, S., Sambridge, M [2006] Quantification of Uncertainty Due to Subgrid Heterogeneity in Reservoir Models, SPE 100223, EUROPEC/EAGE Annual Conference and Exhibition, Vienna, Austria, 12-15 June

Oliver, D.S [1996] Multiple Realizations of the Permeability Field from Well-Test Data, SPE 27970, SPE Journal, volume 1, number 2, 145-154

Oliver, D.S., He, N., Reynolds, A.C [1996] Conditioning Permeability Fields to Pressure Data, In European Conference for the Mathematics of Oil Recovery (ECMOR), Leoben, Austria

Oliver, D.S., Chen, Y [2010] Recent Progress on Reservoir History Matching: A Review, Computational Geoscience, Online First, 19 July

Oliver, K., Stutzle, T., Exner, T [2006] PLANTS, Application of Ant Colony Optimization to Structure-Based Drug Design, In Proceedings of ANTS 2006, Brussels, Belgium 4-7 September

Olsen, G.T., McVay, D.A., Lee, W.J [2005] Inverted Hockey Stick Method Reveals Greater Uncertainty in Petroleum Investment Evaluations, SPE 93999, Western Regional Meeting, Irvine, California, USA, 30 March – 1 April

Omran, M.G.H., Salman, A., Engelbrecht, A.P [2005] Self-Adaptive Differential Evolution, Lecture Notes in Computer Science, 3801, Springer, 192-199

Omran, M.G.H., Engelbrecht, A., Salman, A [2009] Bare Bones Differential Evolution, European Journal of Operational Research, 196, 128-139

Oren, S., Luenberger, D [1974] Self-Scaling Variable Metric Algorithms, Journal of Management Science, volume 20, No 5, 845-862

Ostermeier, A., Gawelczyk, A., Hansen, N [1994] Step-Size Adaption Based on Non-Local Use of Selection Information, In: Davidor, Y., Schwefel, H., Manner, R. (Eds.), Parallel Problem Solving From Nature – PPSN III, LNCS 866, Springer-Verlag, Berlin, Germany, 189-198

O'Sullivan, A., Christie, M [2006] Simulation Error Models for Improved Reservoir Prediction, Reliability Engineering & System Safety, volume 91, issues 10-11, 1382-1389

Ouenes, A., Meunier, G., Moegen, H [1992-A] Application of Simulated Annealing Method (SAM) to Gas Storage Reservoir Characterization, 96th Annual AIChE National Spring Meeting, New Orleans, March 29-April 3

Ouenes, A., Fasanino, G., Lee, R [1992-B] Simulated Annealing for Interpreting Gas/Water Laboratory, SPE 24870, 67th Annual Technical Conference and Exhibition, Washington D.C, USA, October 4-7

Ouenes, A., Brefort, B., Meunier, G., Dupere, S [1993-A] A New Algorithm for Automatic History Matching: Application of Simulated Annealing Method (SAM) to Reservoir Inverse Modeling, SPE 26297, Unsolicited Manuscript

Ouenes, A., Saad, N [1993-B] A New Fast Parallel Simulated Annealing Algorithm for Reservoir Characterization, SPE 26419, SPE Annual Technical Meeting and Exhibition, Houston, Texas, USA, October 4-6

Ouenes, A., Weiss, W., Sultan, A., Anwar, J [1995] Parallel Reservoir Automatic History Matching Using a Network of Workstations and PVM, SPE 29107, Reservoir Simulation Symposium, San Antonio, Texas, USA, 12-15 February

Ouiddir, R., Rahli, M,. Meziane, R., Zeblah, A [2004] Ant Colony Optimization for New Redesign Problem of Multi-State Electrical Power Systems, Journal of Electrical Engineering, 55, 3-4, 57-63

Ouimette J, Oran K [2006] Implementing Chevron *i-field* at the San Ardo, California, SPE 99548, Intelligent Energy Conference and Exhibition, Amsterdam, The Netherlands, 11-13 April

Oye, V., Roth, M [2003] Automated Seismic Event Location for Hydrocarbon Reservoirs, Computers and Geoscience, 29, 851-863

Page, T., Beven, K.J., Whyatt, J.D [2004] Predictive Capability in Estimating Changes in Water Quality: Long-Term Responses to Atmospheric Deposition, Water Air and Soil Pollution, 151, 215-244

Pajonk, O., Krosche, M., Schulze-Riegert, R., Niekamp, R., Matthies, H.G [2008] Stochastic Optimization Using EA and EnKF – A Comparison, 11th European Conference on the Mathematics of Oil Recovery, Bergen, Norway, 8-11 September

Palatnic, B., Zakirov, L., Haugen, S., van Roosmalen, J [1993] New Approaches to Multiple History Matching, to be presented at the Seventh European Symposium on Improved Oil Recovery, Moscow

Pampara, G., Engelbrecht, A.P., Franken, N [2006] Binary Differential Evolution, In IEEE Congress on Evolutionary Computation (CEC 2006), Vancouver, BC, Canada, 1873-1879

Pareto, V [1906] Manuale di Economia Politica, Societa Editrice Libraria, Milano, Italy. Translated into English by A.S. Schwier as Manual of Political Economy, Macmillan, New York, 1971

Parish, R., Calderbank, V., Watkins, A., Muggeridge, A., Goode, T., Robinson, P [1993] Effective History Matching: The Application of Advanced Software Techniques to the History Matching Process, SPE 25250, Reservoir Simulation Symposium, New Orleans, USA, February 28 - March 3

Park, H., Lim, J., Roh, J., Kang, J, Min, B [2006] Production-System Optimization of Gas Fields Using Hybrid Fuzzy-Genetic Approach, SPE 100179, EUROPEC/EAGE Annual Conference and Exhibition, Vienna, Austria, 12-15 June

Parsopoulus, K.E., Taoulis, D.K., Pavilidis, N.G., Plagianakos, V.P., Vrahatis, M.N. [2004] Vector Evaluated Differential Evolution for Multiobjective Optimization, Congress on Evolutionary Computation (CEC 2004), Portland, Oregon, USA

Pawer, P.J., Rao, R.V., Shankar, R [2008] Multiobjective Optimization of Electro-Chemical Machining Process Parameters Using Artificial Bee Colony (ABC) Algorithm, Advances in Mechanical Engineering (AME-2008), Surat, India, December 15-17

Pedersen, S., Randen, T., Sonneland, L., and Steen, O [2002] Automatic 3D Fault Interpretation By Artificial Ants, EAGE 64th Conference and Exhibition, Florence, Italy, 27-30 May

Pena, S., Rionda, B., Aguirre, A.H [2005] Multiobjective Shape Optimization Using Estimation Distribution Algorithms and Correlated Information, Evolutionary Multiobjective Optimization 2005, LNCS 3410, Springer, 664-676

Peng, L., Wang, Y., Dai, G [2008] A Novel Opposition-Based Multi-Objective Differential Evolution Algorithm for Multi-objective Optimization, In: Kang, L (Ed.) ISICA 2008, LNCS 5370, 162-170

Perdue, J [2002] Faster, Better, Deeper, E&P Magazine, February 2002

Petrel [2009] Schlumberger Information Solutions (SIS)

Petrie, R [2008] Localization in the Ensemble Kalman Filter, Department of Meteorology, University of Reading, August

Pham, R., Al-Ajmi, F., Al-Shehab, M [2008] Navigating the Fog of Reservoir Uncertainties to Decision Makings with Advanced Mathematical Models in New Field Development, IPTC 11978, International Petroleum Technology Conference, Kuala Lumpur, Malaysia, 3-5 December

Pickup, G., Valjak, M., Christie, M [2008] Model Complexity in Reservoir Simulation, 11th European Conference on Mathematics of Oil Recovery, Bergen, Norway, 8-11 September

Pourtakdoust, S., Nobahari, H [2004] An Extension of Ant Colony System to Continuous Optimization Problems, In Proceeding of ANTS 2004, Brussels, Belgium, LNCS 3172, 2004, 294-301

Powell, M.J.D [1978] A Fast Algorithm for Nonlinearly Constrained Optimization Calculations, Number 630, Lecture Notes in Mathematics 1066, 144-157, Springer-Verlag, Berlin

Prakash, B.D.K., Shelokar, S., Jayaraman, V.K [2002] Ant Algorithm for Single and Multiobjective Reliability Optimization Problems, Quality and Reliably Engineering, volume 18, number 6, 497-514

Price, K.V., Storn, R., Lampinen, J [2005] Differential Evolution: A Practical Approach to Global Optimization, Springer

Pruess, K., Weres, O., Schroeder, R., Marconcini, R., Neri, G [1980] History Match Simulation of Serrazzano Geothermal Reservoir, SPE 9276, 55th Annual Fall Technical Conference and Exhibition, Dallas, Texas, USA, September 21-24

PUNQ-S3 Model, Department of Earth Science and Engineering, Imperial College London, http://www3.imperial.ac.uk/earthscienceandengineering/research/perm/punq-s3model , Last Accessed May 2010

Purnell, G.W., Nolte, B.J., Krail, P.M., Ebrom, D.A [1999] Analysis of 4-C Data for AVO Effects at Teal South, Eugene Island 354, OTC 10984, Offshore Technology Conference, Houston, Texas, USA, 3-6 May

Qing, A [2009] Differential Evolution: Fundamentals and Applications in Electrical Engineering, ISBN 978-0-470-82392-7, Wiley

Qing, A., Lee, C.K [2010] Differential Evolution in Electromagnetics, ISBN: 978-3-642-12868-4, Springer

Rahman, M.M., Rahman, M.K., Rahman, S.S [2001] An Integrated Model for Multiobjective Design Optimization of Hydraulic Fracturing, Journal of Petroleum Science and Engineering, 31, 41-62

Rajesh, K., Gupta, S., Rangaiah, G., Ray, K [2000] Multiobjective Optimization of Steam Reformer Performance Using Genetic Algorithm, Ind Eng Chem Res, 39, 706-717

Rajpoot, N., Hussain, A., Ali, U., Saleem, K., and Qureshi, M [2004] A Novel Image Coding Algorithm Using Ant Colony System Vector Quantization, International Workshop on Systems, Signals and Image Processing, Poland, September 13-15

Ray, T., Sarker, R [2006] Multiobjective Evolutionary Approach to the Solution of Gas Lift Optimization Problems, IEEE Congress on Evolutionary Computation, Vancouver, Canada, July 16-21

Razavi, F., Jalai-Farahani, F [2008] Ant Colony Optimization: A Leading Algorithm in Future Optimization of Petroleum Engineering Processes, 9th International Conference on Artificial Intelligence and Soft Computing, KAISC 2008, Zakopane, Poland, June 22-26

Rechenberg, I [1965] Cybernetic Solution Path of an Experimental Problem, Royal Aircraft Establishment, Library Translation Number 1122, Farnborough, UK

Resovsky, J., Trampert, J [2002] Reliable Mantle Density Error Bars: An Application of the Neighbourhood Algorithm to Normal-mode and Surface Wave Data, Geophysics Journal, 150, 665-672

REvolution Computing [2009] foreach: Foreach Looping Construct for R, http://cran.r-project.org/web/packages/foreach/index.html

Robert, C [1995] Simulation of Truncated Normal Variables, Statistics and Computing, 5, 121-125

Robic, T., Filipic, B [2005] DEMO: Differential Evolution for Multiobjective Optimization, In: Coello Coello, C.A., Hernandez Aguirre, A., Zitzler, E (Eds.) EMO 2005, Springer LNCS 3410, 520-533

Roche, S., Maxwell, P., Fisseler, G [1999] Teal South 4C/3D Survey: A Model for 4C/4D Seismic Data Acquisition, OTC 10983, Offshore Technology Conference, Houston, Texas, USA, 3-6 May

Romero, C., Carter, J., Zimmerman, R., Gringarten, A [2000-A] Improved Reservoir Characterization Through Evolutionary Computation, SPE 62942, Annual Technical Conference and Exhibition, Dallas, Texas, USA, 1-4 October

Romero, C., Carter, J., Gringarten, A., Zimmerman, R [2000-B] A Modified Genetic Algorithm for Reservoir Characterization, SPE 64765, International Oil and Gas Conference and Exhibition, Beijing, China 7-10 November

Rommelfanger, H [2007] Optimization of Fuzzy Objective Functions in Fuzzy (Multicriteria) Linear Programs – A Critical Survey, In Castillo et al. (Eds.) Theoretical Advances and Applications of Fuzzy Logic and Soft Computing, Advances in Soft Computing, volume 42, 324-33

Ross, T.J [2004] Fuzzy Logic with Engineering Applications, John Wiley

Rotondi, M., Nicotra, G., Godi, A., Contento, F., Blunt, M., Christie, M [2006] Hydrocarbon Production Forecast and Uncertainty Quantification: A Field Application, SPE 102135, Annual Technical Meeting and Exhibition, San Antonio, USA, 24-27 September

Rudduck, N., Khurana, A., Congreve, M., Lourens, J [2006] Multi-Objective Decision Making: A Critical Analysis of the Applicability of Renewable Energy Technologies, SPE 101770, Asia Pacific Oil and Gas Conference and Exhibition, Adelaide, Australia, 11-13 September

Ryu, J.H., Kim, S., Wan, H [2009] Pareto Front Approximation with Adaptive Weighted Sum Method in Multiobjective Simulation Optimization, Proceedings of the 2009 Winter Simulation Conference, Austin, Texas, USA

Saad, G., Ghanem, R [2009] Characterization of Reservoir Simulation Models Using a Polynomial Chaos-Based Ensemble Kalman Filter, Water Resources Research, volume 45, W04417

Sablok, R., Aziz, K [2008] Upscaling and Discretization Errors in Reservoir Simulation, Petroleum Science and Technology, volume 26, issue 10 & 11, 1161-1186

Sadeghzadeh, K [2009] Multiobjective Optimization in Variably Saturated Fluid Flow, Journal of Computational and Applied Mathematics, 223, 801-819

Sahni, A [2003] Case Studies of Uncertainty Analysis in the Seismic to Reservoir Simulation Workflow, SPE 84188, Annual Technical Conference and Exhibition, Denver, Colorado, USA, 5-8 October

Sakawa, M [1982] Interactive Multiobjective Decision Making by the Sequential Proxy Optimization Technique, European Journal of Operational Research, 9, 386-396

Saleh, H [2002] Ants Can Successfully Design GPS Surveying Networks, GPS World, September issue, 48-60

Salhi, A., Toreyen, O [2010] A Game Theory-Based Multi-Agent System for Expensive Optimization Problems, In Tenne, Y and Goh, C.K (Eds.) Computational Intelligence in Optimization, ALO 7, Springer, 211-232

Saleri, N., Toronyl, R., Snyder, D [1992] Data and Data Hierarchy, SPE 21369, Journal of Petroleum Technology, volume 44, number 12, 1286-1293

Sambridge, M [1999 a] Geophysical Inversion with a Neighbourhood Algorithm, Part 1: Searching Parameter Space, Geophysical Journal International, 138, 479-494

Sambridge, M [1999 b] Geophysical Inversion with a Neighbourhood Algorithm - II Appraising the Ensemble, Geophysics Journal Int, 138, 727-745

Sambridge, M [2003] Nonlinear Inversion by Direct Search Using the Neighbourhood Algorithm, International Handbook of Earthquakes and Engineering Seismology, volume 81B, 35-37

Sandgren, E [1990] Nonlinear Integer and Discrete Programming in Mechanical Design Optimization, Journal of Mechanical Design, 112, 223-229

Sankaran, A., Lugo, J., Awashti, A., Mijares, G [2009] The Promise and Challenges of Digital Oilfield Solutions – Lessons Learned From Global Implementations and Future Directions, SPE 122855, Digital Energy Conference & Exhibition, Houston, Texas, USA, 7-8 April

Sarich, M [2001] Using Genetic Algorithms to Improve Investment Decision Making, SPE 68725, Asia Pacific Oil and Gas Conference and Exhibition, Jakarta, Indonesia, 17-19 April

Sarma, P., Chen, W [2009] Generalization of the Ensemble Kalman Filter Using Kernels for Non-Gaussian Random Fields, SPE 119177, Reservoir Simulation Symposium, The Woodlands, Texas, USA, 2-4 February

Savoia, M., Vincenzi, L [2008] Differential Evolution Algorithm for Dynamic Structural Identification, Journal of Earthquake Engineering, volume 12, issue 5, 800-821

Schluter, M., Egea, J., Antelo, L., Alonso, A., Banga, R [2009] An Extended Ant Colony Optimization Algorithm for Integrated Process and Control System Design, Ind. Eng. Chem. Res, 48, 6723-6738

Schmidt, H. and Thierauf, G [2005] A Combined Heuristic Optimization Technique, Advances in Engineering Software, 36, 11-19

Schulze-Riegert, R., Axmann, J., Haase, O., Rian, D., You, Y [2001] Optimization Methods for History Matching of Complex Reservoirs, SPE 66393, Reservoir Simulation Symposium, Houston, Texas, USA, 11-14 February

Schulze-Riegert, R., Krosche, M ., Fahimuddin, A., Ghedan, S [2007] Multiobjective Optimization with Application to Model Validation and Uncertainty Quantification, SPE 105313, 15th Middle East Oil and Gas Show and Conference, Bahrain, 11-14 March

Schulze-Riegert, R., Krosche, M., Pajonk, O., Mustafa, H [2009] Data Assimilation Coupled to Evolutionary Algorithms – A Case Example in History Matching, SPE 125512, SPE/EAGE Reservoir Characterization and Simulation Conference, Abu Dhabi, UAE, 19-21 October

Schwefel, H [1968] MHD Ramjet Project: Experimental Optimization of Two Phase Injector, Technical Report 11.034/68, AEG Forschungsinstitut, Berlin

Sedighi, F., Stephen, K [2009] Faster Convergence in Seismic History Matching by Efficient Parameter Searching, SPE 121210, EUROPEC/EAGE Annual Conference and Exhibition, Amsterdam, The Netherlands, 8-11 June

Selberg, S., Schulze-Riegert, R., Stekolschikov, K [2007] Event-Targeting Model Calibration Used for History Matching Large Simulation Cases, SPE 106044, Reservoir Simulation Symposium, Houston, Texas, USA, 26-28 February

Sen, M., Datta-Gupta, A., Stoffa, P., Lake, L., Pope, G [1995] Stochastic Reservoir Modeling Using Simulated Annealing and Genetic Algorithms, SPE 24754, SPE Formation Evaluation, volume 10, number 1, 49-55

Seo, J., Shneiderman, B [2005] A Knowledge Integration Framework for Information Visualization, In Hemmje M, et al (Eds) From Integrated Publication and Information Systems to Virtual Information and Knowledge Environments, Lecture Notes in Computer Science, 3379, Springer, 207-220

Serbini, F., Wee, L.K., Wong, L.H, Gomez, N [2009] Integrated Field Development – Improved Field Planning and Operation Optimization, IPTC 14010, International Petroleum Technology Conference, Doha, Qatar, 7-9 December

Shafii, M., De Smedt, F [2009] Multiobjective Calibration of a Distributed Hydrological Model (WetSpa) Using a Genetic Algorithm, Hydrol Earth Syst Sci, 13, 2137-2149

Shah, P., Gavalas, G., Seinfeld, J [1978] Error Analysis in History Matching: The Optimum Level of Parameterization, SPE Journal, volume 18, number 3, 219-228

Shah Hosseini, H [2007] Problem Solving by Intelligent Water Drops, IEEE Congress on Evolutionary Computation, CEC 2007, Singapore, 3226-3231

Sharma, A., Leung, J., Srinivasan, S., Kim, Y [2008] An Integrated Approach to Reservoir Uncertainty Assessment, Case Study of a Gulf of Mexico Reservoir, SPE 116351, Annual Technical Conference and Exhibition, Denver, Colorado, USA, 21-24 September

Sharma, S [2009] Application of Parallel Genetic Algorithms in Ground Water Contamination Problems: An Overview, In proceedings of World Congress on Nature and Biologically Inspired Computing, 1681-1685

Shelokar, P., Siarry, P., Jayaraman, V., Kulkarni, B [2007] Particle Swarm and Ant Colony Algorithms Hybridized for Improved Continuous Optimization, Applied Mathematics and Computation, 188, 129-142

Shenoy, P [2006] Inference in Hybrid Bayesian Networks Using Mixtures of Gaussian, In Dechter, R and Richardson, T (Eds.) Uncertainty in Artificial Intelligence Proceedings of the Twenty-Second Conference (UAI-06), AUIA Press, Corvallis, OR, 428-436

Siarry, P., Berthiau, G., Durbin, F., Haussy, J [1997] Enhanced Simulated Annealing for Globally Minimizing Functions of Many Continuous Variables, ACM Transactions on Mathematical Software 23 (2), 209-228

SIM Reservoir, Kongsberg Oil and Gas, Norway, Last Accessed November 2010
http://www.kongsberg.com/en/KOGT/Products/SimReservoir.aspx

Slater, G., Durrer, E [1970] Adjustment of Reservoir Simulation Models to Match Field Performance, SPE 2983, 45th Annul Fall Meeting, Houston, Texas, USA, 4-7 October

Socha, K [2008] Ant Colony Optimization for Continuous and Mixed Variable Domains, PhD Thesis, IRIDIA-CoDE, University of Brussels

Socha, K., Dorigo, M [2008] Ant Colony Optimization for Continuous Domains, European Journal of Operational Research, volume 185, number 3, 1155-1173

Soleng, H [1999] Oil Reservoir Production Forecasting with Uncertainty Estimation Using Genetic Algorithms, In IEEE Evolutionary Computation Conference (CEC), Washington D.C., USA

Sousa, S., Maschio, C., Schiozer, D [2006] Scatter Search Metaheuristic Applied to the History-Matching Problem, SPE 102975, Annual Technical Conference and Exhibition, San Antonio, Texas, USA, 24-27 September

Srinivas, N., Deb, K [1994] Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms, Evolutionary Computation, volume 2, issue 3, 221-248

Steck, H., Jaakkola, T [2004] Bias-Corrected Bootstrap and Model Uncertainty, In Advances in Neural Information Processing Systems, 16, MIT Press

Stedinger, J.R., Vogel, R.M., Lee, S.U., Batchelder, R [2008] Appraisal of the Generalized Likelihood Uncertainty Estimation (GLUE) Method, Water Recourses Research, volume 44, W00B06

Steffen, H [2006] Determination of a Consistent Viscosity Distribution in the Earth's Mantle Beneath Northern and Central Europe, PhD Thesis, Institute of Geology, Free University of Berlin, Germany

Stephen, K., MacBeth, C [2006] Reducing Reservoir Prediction Uncertainty Using Seismic History Matching, SPE 100295, EUROPEC/EAGE Annual Conference and Exhibition, Vienna, Austria, 12-15 June

Stephen, K., Shams, A., MacBeth, C [2007] Faster Seismic History Matching in a UKCS Reservoir, SPE 107147, EUROPEC/EAGE Annual Conference and Exhibition, London, United Kingdom, 11-14 June

Steuer, R.E [1986] Multiple Criteria Optimization: Theory, Computation, and Application, Wiley, Chichester

Stewart, T., Bandte, O., Braun, H., Chakraborti, N., Ehrgott, M., Gobelt, M., Jin, Y., Nakayama, H., Poles, S., Di Stefano, D [2008] Real-World Application of Multiobjective Optimization, In Branke, J et al. (Eds.) Multiobjective Optimization, LNCS 5252, Springer, 285-327

Storn, R., Price, K [1995] Differential Evolution – A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces, Technical Report for International Computer Science Institute, Berkeley, TR-95-012

Storn, R. Price, K [1997] Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces, Journal of Global Optimization, 11, 341-359

Stutzle, T [1998] Parallelization Strategies for Ant Colony Optimization, Proceedings of the Fifths International Conference on Parallel Problem Solving from Nature, 722-731

Subbey, S., Christie, M., Sambridge, M [2003] A Strategy for Rapid Quantification of Uncertainty in Reservoir Performance Prediction, SPE 79678, Reservoir Simulation Symposium, Houston, USA, 3-5 February

Sultan, A., Ouenes, A., Weiss, W [1993] Reservoir Description by Inverse Modeling: Application to EVGSAU Field, SPE 26478, 68th Annual Technical Meeting Conference and Exhibition, Houston, Texas, USA, 3-5 October

Sultanum, N.B., Sharlin, E., Sousa, M.C., Miranda-Filho, D.N., Eastick, R [2010] Touching the Depths: Introducing Tabletop Interactions to Reservoir Engineering, ACM International Conference on Interactive Tabletops and Surfaces (ITS), Saarbrucken, Germany, 7-10 November

Suribabu, C.R [2010] Differential Evolution Algorithm for Optimal Design of Water Distribution Networks, Journal of Hydroinformatics, volume 12, number 1, 66-82

Suzuki, S [2007] Integrated Evaluation of Structural Uncertainty Using History Matching from Seismic Imaging Uncertainty Model, PhD Thesis, Stanford University

T'kindt, V., Monmarche, N., Tercinet, F., Laugt, D [2002] Ant Colony Optimization Algorithm to Solve a 2-Machine Bicriteria Flowshop Scheduling Problem, European Journal of Operational Research, volume 142, number 2, 250-257

Talbi, E., Mostaghim, S., Okabe, T., Ishibuchi, H., Rudolph, G., Coello, C [2008] Parallel Approaches for Multiobjective Optimization, In Branke et al (Eds.) Multiobjective Optimization, LNCS 5252, 349-372

Tan, K.C., Khor, E.F., Lee, T.H [2005] Multiobjective Evolutionary Algorithms and Applications, Springer

Tan, T [1995] A Computationally Efficient Gauss-Newton Method for Automatic History Matching, SPE 29100, Reservoir Simulation Symposium, San Antonio, Texas, USA, 12-15 February 1995

Tavassoli, Z., Carter, J., King, P [2004] Errors in History Matching, SPE 86883, SPE Journal, volume 9, number 3, 352-361

Teo, J [2005] Exploring Dynamic Self-Adaptive Populations in Differential Evolution, Soft Computing – A Fusion Foundations, Methodologies and Applications, DOI 10.1007/s05500-055-0537-1

Tfaili, W., Siarry, P [2008] A New Charged Ant Colony Algorithm for Continuous Dynamic Optimization, Applied Mathematics and Computation, 197, 604-613

Thierauf, G., Cai, J [2000] Evolution Strategies-Parallelization and Application in Engineering Optimization, In B. Topping (Ed.), Parallel and Distributed Processing for Computational Mechanics: Systems and Tools, Saxe-Coburgh Publications, Edinburgh, UK, 329-349

Thomas, K., Hellums, L., Reheis, G [1971] A Nonlinear Automatic History Matching Technique for Reservoir Simulation Models, SPE 3475, 46th Annual Fall Meeting, New Orleans, USA, October 3-6

Thomson A [2008] Deploying FIELD OF THE FUTURE on Major Projects, SPE 112117, Intelligent Energy Conference and Exhibition, Amsterdam, The Netherlands, 25-27 February

Tierney, L., Rossini, A.J., Li, N., Sevcikova, H [2008] SNOW: Simple Network of Workstations, http://cran.r-project.org/web/packages/snow/index.html

Timms, J., Neal, M [2009] Investigating the Evolution and Stability of a Resource Limited Artificial Immune System, Proceedings of the Genetic and Evolutionary Computation Conference, Workshop on Artificial Immune Systems, 40-41

Tizhoosh, H.R [2005] Opposition-Based Learning: A New Scheme for Machine Intelligence, In Proc Int Conf Compu Intell, Modeling Control and Autom, Vienna, Austria, volume 1, 695-701

Tukey, J [1977] Exploratory Data Analysis, Addison-Wesley

Tupac, Y., Faletti, L., Pacheco, M., Vellasco, M [2007] Evolutionary Optimization of Oilfield Development, SPE 107552, Digital Energy Conference and Exhibition, Houston, Texas, USA, 11-12 April

Tvrdík, J [2006] Competitive Differential Evolution and Genetic Algorithm in GA-DS Toolbox, In Technical Computing Prague, Praha, Humusoft, Full version at http://dsp.vscht.cz/konference_matlab/MATLAB06/prispevky/tvrdik/tvrdik.pdf

Twartz, S.K., Gorjy, F., Milne, I.G [1998] A Multiple Realization Approach to Managing Uncertainty in the North Rankin Gas Condensate Field, Western Australia, SPE 50078, Asia Pacific Oil and Gas Conference and Exhibition, Perth, Australia, 12-14 October

Tyler, K., Svanes, T., Omdal, S [1993] Faster History Matching and Uncertainty in Predicted Production Profiles with Stochastic Modeling, SPE 26420, 68th Annual Technical Conference and Exhibition of the SPE, Houston, Texas, USA, 3-6 October

Ulrich, T., Bader, J., Zitzler, E [2010] Integrating Decision Space Diversity into Hypervolume-Based Multiobjective Search, In Genetic and Evolutionary Computation Conference (GECCO 2010) New York, NY, USA

Ursem, R. Vadstrup, P [2003] Parameter Identification of Induction Motors Using Differential Evolution, in Evolutionary Computation, Canberra, Australia, 790-796

Vallee, M., Bouchon, M [2004] Imaging Coseismic Rupture in Far Field by Slip Patches, Geophysics Journal, volume 156, issue 3, 615-630

Valles, B., Naevdal, G [2008] Comparing Different Ensemble Kalman Filter Approaches, 11th European Conference on the Mathematics of Oil Recovery, Bergen, Norway, 8-11 September

van Elk, J.F., Guerrera, L., Vijayan, K., Gupta, R [2000] Improved Uncertainty Management in Field Development Studies Through Application of the Experimental Design Method to the Multiple Realization Approach, SPE 64462, Asia Pacific Oil and Gas Conference and Exhibition, Brisbane, Australia, 16-18 October

van Essen, G.M., Van den. H., Jansen, J.D [2009] Hierarchical Long-Term and Short-Term Production Optimization, SPE 124332, Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 4-7 October

Veldhuizen, D.V., Lamont, G.B [2000] Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art, Evolutionary Computation Journal, volume 8, number 2, 125-148

Vesterstrom, J. Thomson, R. A [2004] Comparative Study of Differential Evolution, Particle Swarm Optimization and Evolutionary Algorithm on Numerical Benchmark Problems, Evolutionary Computation, IEEE, Portland, Oregon, USA

Volpi, B., Aliverti, E., Astorino, G., Botta, M., Calzari, M., Contento, F.M., Piantanida, M., Tarchiani, C [2008] From Reservoir Studies Best Practices to an Advanced Web Based System to Set Up and Monitor Project Activities, 70th EAGE Conference and Exhibition, Rome, Italy, 9-12 June

Walker, G., Kromah, M., Pham, H., Adeyeye, D., Winchester, A [2008] Assisted History Matching as a Useful Tool: An Example from Trinidad, SPE 113479, Indian Oil and Gas Technical Conference, Mumbai, India, 4-6 March

Walker, G., Pettigrew, S [2006] Measure of Efficiency for Assisted History Matching, paper A-005, 10th European Conference on the Mathematics of Oil Recovery, Amsterdam, The Netherlands, 4-7 September 2006

Walstrom, J., Mueller, T., McFarlane, R [1967] Evaluating Uncertainty in Engineering Calculations, SPE 1928, 42nd Annual Fall Meeting, Houston, Texas, USA, 1-4 October

Wan, Z., Igusa, T [2003] Adaptive Sampling for Optimization Under Uncertainty, Proceedings of the Fourth International Symposium on Uncertainty Modeling and Analysis (ISUMA), IEEE

Wang, J., Buckley, J.S [2006] Automatic History Matching Using Differential Evolution Algorithm, International Symposium of the Society of Core Analysts, Trondheim, Norway, 12-16 September

Wang, J., Eseosa, O., Thulasiraman, P [2009] HOPNET: A Hybrid Ant Colony Optimization Routing Algorithm for Mobile Ad Hoc Networks, Ad Hoc Networks, 7, 690-705

Wang, K., Ye, M [2009] Parameter Determination of Schottky Barrier Diode Model Using Differential Evolution, Solid-State Electronics, 53, 234-240

Wang, P [2003] Development and Applications of Production Optimization Techniques for Petroleum Fields, PhD Thesis, Stanford University

Wathelet, M [2005] Array Recordings of Ambient Vibrations, PhD Thesis, Leige University, Belgium

Wathelet, M [2008] An Improved Neighbourhood Algorithm: Parameter Conditions and Dynamic Scaling, Geophysical Research Letters, volume 35, L09301

Watkins, A., Parish, R ., Modine, A [1992] A Stochastic Role for Reservoir Engineering Input to Reservoir History Matching, SPE 23738, Second Latin American Petroleum Engineering Conference, Caracas, Venezuela, March 8-11

Watson, A., Lee, W [1986] A New Algorithm for Automatic History Matching Production Data, SPE 15228, Unconventional Gas Technology Symposium, Louisville, USA, May 18-21

Weber, E.U., Johnson, E.J [2008] Decisions Under Uncertainty: Psychological, Economic, and Neuroeconomic Explanations of Risk Preference, In Glimcher, P.W (Ed.) Neuroeconomic Decision Making and the Brain, Elsevier

Welsh, M., Lee, M., Begg, S.H [2009] Repeated Judgments in Elicitation Tasks: Efficacy of the MOLE Methods, in N.A. Taatgen & H. van Rijn (Eds.), Proceedings of

the 31st Annual Conference of the Cognitive Science Society, Austin, Texas, USA, 1529-1534

Wetter, M., Wright, J [2004] A Comparison of Deterministic and Probabilistic Optimization Algorithms for NonSmooth Simulation-Based Optimization, Building and Environment, 39, 989-999

Wierzbicki, A.P., Makowski, M., Wessels, J (Eds.) [2000] Decision Support Methodology with Environmental Application, Kluwer Academic Publishers, Dordrecht

Williams, G., Mansfield, M., MacDonald, D., Bush, M [2004] Top-down Reservoir Modeling, SPE 89974, Annual Technical Conference and Exhibition, Houston, Texas, USA, 26-29 September

Williams, N., Lond, D [2006] Kutubu – A Rethink, SPE 101123, Asia Pacific Oil and Gas Conference and Exhibition, Adelaide, Australia, 11-13 September

Williams, G [2010] Personal Communication and Discussion, Steering Group Meeting of Uncertainty Project, Heriot Watt University, Edinburgh

Wising, U., Kalitventzeff, P., Campan, J., Vrielynck, B [2009] Improving Operations Through Increased Accuracy of Production Data, SPE 124766, Offshore Europe Conference, Aberdeen, UK, 8-11 September

Witt, A., Barr, D., Fowler, S., McGarrity, J., Towns, M., Kjelstadli, R [2010] Managing the Start-up of a Fractured Oil Reservoir: Development of the Clair Field, West of Shetland, AAPG Student Talk, Heriot Watt University, 21 July

Wodrich M [1996] Ant Colony Optimization, BSc Thesis, Department of Electrical and Electronic Engineering, University of Cape Town, South Africa

Wong, P., Aminzadeh, F., Nikravesh, M [2002] Soft Computing for Reservoir Characterization and Modeling, Physica-Verlag, Springer

Wu, C., Tseng, K [2010] Stress-Based Binary Differential Evolution for Topology Optimization of Structures, Journal of Mechanical Engineering Science, volume 224, number 2, 443-457

Wu, S. and Chow, P [1995] Genetic Algorithm for Nonlinear Mixed Discrete Integer Optimization Problems via Meta-Genetic Parameter Optimization, Engineering Optimization, volume 23, number 2, 137-159

Xu, X., Li, Y [2007] Comparison Between Particle Swarm Optimization, Differential Evolution and Multi-Parents Crossover, Proceedings of the International Conference on Computational Intelligence and Security, IEEE, 124-127

Xue, F., Sanderson, A.C, Bonissone, P.P., Graves, R [2005] Fuzzy Logic Controlled Multi-Objective Differential Evolution, 14th IEEE International Conference on Fuzzy Systems, 25-27 May, 720-725

Xue, F., Sanderson, A.C., Graves, R.J [2003] Pareto-based Multiobjective Differential Evolution Algorithm, In Proceedings of the 2003 Congress on Evolutionary Computation (CEC 2003), Canberra, Australia, volume 2, 862-869

Yamada, T [2000] Non-Uniqueness of History Matching, SPE 59434, Asia Pacific Conference on Integrated Modeling for Asset Management, Yokohama, Japan, 25-26 April

Yang, J.B [1999] Gradient Projection and Local Region Search for Multiobjective Optimization, European Journal of Operational Research, 112, 432-459

Yang, P., Watson, A [1988] Automatic History Matching with Variable-Metric Methods, SPE 16977, SPE Reservoir Engineering, volume 3, number 3, 995-1001

Yao, H., Beghein, C., van der Hilst, R [2009] Surface Wave Array Tomography in SE Tibet from Ambient Seismic Noise and Two-Station Analysis – II. Crustal and Upper-Manle Structure, Geophysics Journal, 173, 205-219

Yapo, P., Gupta, H., Sorooshian, S [1997] Multi-Objective Global Optimization for Hydrologic Model, Journal of Hydrology, 204, 83-97

Yoshizawa, K., Kennett, B [2002] Non-linear Waveform Inversion for Surface Waves with a Neighbourhood Algorithm – Application to Multimode Dispersion Measurements, Geophysics Journal, 149, 118-133

Yu, H [2010] Rmpi: Interface to MPI (Message-Passing Interface), http://cran.r-project.org/web/packages/Rmpi/index.html

Yuan, C., Druzdzel, M.J [2007] Importance Sampling for General Hybrid Bayesian Networkds, Eleventh International Conference on Artificial Intelligence and Statistics, San Juan, Puerto Rico, 21-24 March

Zadeh, L [1965] Fuzzy Sets, Information and Control, volume 8, 338-353

Zadeh, L.A [1978] Fuzzy Sets as a Basis for a Theory of Possibility, Fuzzy Sets and Systems, 1, 3.28

Zaharie, D [2007] A Comparative Analysis of Crossover Variants in Differential Evolution, In Proceedings of the International Multiconference on Computer Science and Information Technology (IMCSIT), Wisla, Poland, 171-181

Zaharie, D [2009] Influence of Crossover on the Behavior of Differential Evolution Algorithms, Applied Soft Computing, 9, 1126-1138

Zamuda, A., Brest, J., Boskovic, B., Zumer, V [2007] Differential Evolution for Multiobjective Optimization with Self Adaption, IEEE Congress on Evolutionary Computation (CEC 2007), 3617-3624

Zamuda, A., Brest, J., Boskovic, B., Zumer, V [2009] Differential Evolution with Self-Adaption and Local Search for Constrained Multiobjective Optimization, Congress on Evolutionary Computation, (CEC 2009), 195-202

Zhai, Z., Zaki, K., Marinello, S., Abou-Sated, A [2009] Coupled Thermo-Poro-Mechanical Effects on Borehole Stability, SPE 123427, Annual Technical Conference and Exhibition, New Orleans, Louisiana, 4-7 October

Zhang, B., Chen, D., Zhao, W [2005] Iterative Ant Colony Algorithm and Its Application to Dynamic Optimization of Chemical Process, Computers and Chemical Engineering, 29, 2078-2086

Zhang, J., Chen, W.N., Zhong, J.H., Tan, X., Li, Y [2006] Continuous Function Optimization Using Hybrid Ant Colony Approach with Orthogonal Design Scheme, Proceedings of SEAL 2006, Springer LNCS 4247, 126-133

Zhang, Q., Lai, L., Wei, H [2007] Continuous Space Optimized Artificial Ant Colony for Real-time Typhoon Eye Tracking, Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Montréal, Canada, 7-10 October

Zhang, D., Gong, X., Peng, L [2009] Estimating Geostatistics Variogram Parameters Based on Hybrid Orthogonal Differential Evolution Algorithm, In Cai et al. (Eds.) 4th International Symposium on Intelligence Computation and Applications, ISICA 2009 Conference, Huangshi, China, October 23-25

Zhang, J., Sanderson, A.C [2009] Adaptive Differential Evolution, ALO 1, 95-113, Springer-Verlag Berlin Heidelberg

Zhang, G [2010] Quantum-Inspired Evolutionary Algorithms: A Survey and Empirical Study, Journal of Heuristics, Online First, Pages 1-49

Zhou, L., Mardambek, J [2008] Reservoir Fluid Evaluation from Real Time Pressure Gradient Analysis: Discussions on Principles, Workflows and Applications, SPE 115045, Asia Pacific Oil and Gas Conference and Exhibition, Perth, Australia, 20-22 October

Zhou, J., Li, Y., Liu, J [2008] Simulation and Optimization of Continuous Pig Lift Systems, PETSOC 08-12-55, Journal of Canadian Petroleum Technology, volume 47, number 12, 55-60

Zhu, W [2010] Massively Parallel Differential Evolution – Pattern Search Optimization with Graphics Hardware Acceleration: An Investigation on Bound Constrained Optimization Problem, Journal of Global Optimization, Online First Article

Zionts, S., Wallenius, J [1976] An Interactive Programming Method for Solving the Multiple Criteria Problem, Management Science, 22, 652-663

Zitzler, E., Deb, K., Thiele, L [2000] Comparison of Multiobjective Evolutionary Algorithms: Empirical Results, Evolutionary Computation Journal, volume 8, number 2, 173-195

Zubarev, D.I [2009] Pros and Cons of Applying Proxy-Models as a Substitute for Full Reservoir Simulations, SPE 124815, Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 4-7 October

Zuber, M., Sawyer, W., Schraufnagel, R., Kuuskaa, V [1987] The Use of Simulation and History Matching Determine Critical Coalbed Methane Reservoir Properties, SPE 16420, SPE/DOE Low Permeability Reservoirs Symposium, Denver, Colorado, USA, May 16-19